Los principales de Elektor

Los 15 mejores artículos de Elektor







Índice de contenidos



ElekTrack

Localización con GPS

Chris Vossen

Localizar está de moda. El archiconocido sistema de navegación TomTom incluso cotiza en bolsa. Asimismo, los sistemas de alarma de los coches más caros y otros vehículos incorporan a menudo de un sistema de este tipo, para así poder saber donde se encuentra el vehículo. Por desgracia, esto último es bastante caro. Así que nos pusimos manos a la obra y hemos desarrollado nuestro propio sistema, el ElekTrack.

Hoy en día queremos saber la ubicación exacta de cualquier cosa. ¿Dónde está el paquete que encargué? ¿Este libro ha vuelto ya a la biblioteca? ¿Habrá llegado ya el tren de mi sobrino? La técnica ha avanzado tanto en los últimos años, que hay realmente una solución para todo, por raro que sea. ¡Y si no, nos encargamos de encontrarla!

Gran Hermano

La gente suele ser un poco reacia ante los organismos que lo tienen todo bajo control. Pero en algunos casos esto es precisamente lo que desean. Nos referimos a la seguridad o a los sistemas de alarma. Los pocos de entre nosotros que podemos permitirnos el nuevo Maybach o un Mercedes SLR, seguro que en el momento de asegurar estos bólidos, lo haremos con estrictas condiciones de seguridad. Por regla general, estos vehículos deben ir equipados con un sistema de alarma de cómo mínimo clase 4 (para una breve explicación, véase [1]). Esto significa que lleva incorporado un sistema de localización basado en GPS, que transmite a una central de avisos el lugar donde se encuentra el vehículo. Los vehículos robados aparecen en el mapa de la sala de control con una lucecita roja.

No sólo los turismos están provistos de un sistema de rastreo. También los camiones y los barcos pueden sacar provecho de estos sistemas. Pero incluso las excavadoras disponen cada vez más de este sistema esta seguridad. Las roban sobre todo en los bosques. Y con un módulo de localización y seguimiento, se pueden localizar estas costosas máquinas.

En fin, no todo el mundo tiene una excavadora o un Bentley en el garaje. Pero esto no significa que un localizador GPS no pueda tener otras aplicaciones. Por ejemplo, en un scooter o una motocicleta. Y si usted es detective privado, seguro que se le hace la boca agua ante un sistema de estos. Para ofrecerle la posibilidad de poder también disfrutar en persona de un localizador GPS, hemos desarrollado el ElekTrack. Debido al gran número de componentes SMD y las dificultades que suelen aparecer al soldarlos, hemos decidido entregar este módulo totalmente montado.

Propósito del proyecto

¿Qué debe poder hacer un soporte de localización en su forma más simple? Está claro que querremos saber donde se encuentra el módulo. Para la parte de localización hemos elegido la solución más a mano: GPS. Así el sistema sabrá en casi todo el mundo donde se encuentra. Además, actualmente el GPS es muy preciso. Y mientras no esté operativo el Galileo europeo [2], es la mejor "alternativa".

Para la transmisión de datos hemos elegido el sistema de SMS. Si bien es cierto que la transmisión de datos no es realmente en directo, es posible casi siempre y en todas partes. Otra buena opción para la transmisión de datos, habría sido la red

Figura 1. En el esquema se puede ver fácilmente que todo está montado alrededor del módem GSM.



GPRS. GPRS funciona con una conexión directa a través de Internet, con lo cual se pueden mostrar los datos en directo a través de un ordenador con acceso a Internet. De este modo, es posible pedir la localización sin retrasos.

Con esto, los datos también se pueden guardar "en directo" en una base de datos. Sin embargo, dimos preferencia al sistema SMS, que es más simple. Por tanto, usaremos dos módulos importantes: un módulo GPS y un módulo GSM.

El módulo GSM es de Wavecom, el Q2686. Este módulo se puede controlar con el código OpenAT de Wavecom y viene equipado con un microprocesador. Esto nos ofrece la gran ventaja de que el proceso de diseño es más corto en comparación con un módem suelto. Pues así no hace falta instalar ningún microcontrolador con eventual periferia externa. Todo se encuentra en un mismo módulo. Y el diseño completo es mucho más compacto.

Para el módulo GPS hemos elegido el Copernicus de Trimble. Es un sucesor del Lassen iQ, el módulo que ya ha demostrado sus cualidades en el receptor GPS para USB (Elektor de julio de 2005).

Esquema

Un vistazo al esquema (**figura 1**) permite ver rápidamente que todo está montado alrededor del módem GSM. El módulo GPS se conecta a través de un convertidor de nivel lógico (IC3) en UART2 del módem. Alrededor de T3, T4 y T5 se ha montado un circuito que detecta si el circuito de antena está puesto en cortocircuito, abierto o conectado normalmente. El módulo GPS



proporciona tensión de suministro a la antena activa a través de T1 y T2.

Para que este circuito funcione bien se necesitan dos tensiones de alimentación, 4,3 y 3,3 voltios. Para ello utilizamos IC5 e IC6. El IC5 (un LT3430) es un convertidor buck que funciona a una frecuencia de 200 kHz. Las resistencias R12 y R14 determinan el ajuste de la tensión de salida. Se ha ajusta aproximadamente a 4,5 voltios.
IC6 se encarga de la tensión de suministro del módulo GPS. Con una tensión de entrada de 4,3 voltios, ofrece una tensión de salida fija de 3,3 voltios. Alrededor de T6, T7 y T8 se han montado tres convertidores de nivel idénticos que transforman las elevadas tensiones de entrada en valores lógicos comprendidos en la especificación del módem GSM.

IC1 es un convertidor RS232 clásico, con el cual se puede hacer, entre otras cosas, una conexión con el PC para poder programar el módem. También es posible hacer una conexión a través de IC1 para aplicaciones propias.
D1 y D2 protegen la entrada del módulo GSM de cargas estáticas que pueden producirse cuando se inserta la tarjeta SIM en la ranura.
El Led D3 muestra el estado de la conexión GSM. Si está continuamente iluminado, entonces el módem no está registrado en la red. En cuanto empieza a parpadear, el módem se ha podido registrar con éxito. El led D12 muestra el estado del módulo GSM.

Conexiones

La conexión de alimentación puede ser a 12 voltios o a 24 voltios. Tras las pruebas, se concluyó que el ElekTrack necesita una tensión mínima de 8 voltios para poder funcionar bien. Asimismo, la alimentación debe poder suministrar suficiente corriente. Cuando el módem GSM está en proceso de transmisión de datos, pueden intervenir picos de corriente relativamente altos. El módulo consume de media unos 500 mW. Por tanto, una batería de 9V sólo durará unas dos horas. Una batería de ciclomotor o coche aguantará más tiempo. Naturalmente, también se pueden incorporar adaptaciones al software para reducir el consumo de energía. Por ejemplo, se puede configurar para que el módulo GPS se coloque en modo "ahorro de energía" con un determinado mensaje SMS y que se reactive con otra orden.

Por supuesto, el módulo también tiene conexiones para antenas. Una para el módulo GPS y una para el módulo GSM. El módem también dispone de varias entradas lógicas, que se pueden utilizar, por ejemplo, para conectar una alarma al ElekTrack. Una tensión superior a unos 8 voltios provocará la activación de la entrada digital del módem GSM. También en ese momento se puede enviar un SMS a un número de teléfono previamente programado. Esto debe instalarlo usted mismo en el software, (todavía) no viene instalado de serie. Es probable que se incluya en una actualización de firmware.
El módulo GPS se puede configurar con la ayuda de varios comandos. Para ello, debe conectarse el módulo al PC con el cable de serie. Entonces será necesario un programa como Hyperterminal para Windows para configurar el ElekTrack. El puerto de serie del PC debe configurarse como en la **figura 2**.

Inicio

En primer lugar, debe comprobarse si la tarjeta SIM está protegida con un código pin. Se puede hacer con el comando `at+cpin?`
Si el módem responde con '`+CPIN: READY`', , significa que la tarjeta sim no necesita ningún código pin. Pero si la respuesta es '`+CPIN:SIM PIN`', entonces la tarjeta sim tiene un código pin y deberá eliminarse.
Por el momento, el firmware todavía no soporta el uso automático del código pin. Probablemente lo haga un futuro firmware. Compruebe si hay nuevas versiones en nuestro sitio web.

Cuando introducimos el comando `at+cpin=0000 0000` (siendo 0000 el código pin de la tarjeta SIM), nos contesta 'OK'. Seguidamente, debe eliminarse la pregunta del código pin.

Esto se hace de la siguiente manera:
`at+clck="SC",0,0000`

(0000 vuelve a ser el código pin de la tarjeta SIM). El módem vuelve indicar de nuevo 'OK'.

Para comprobar que el módem se registra automáticamente en la red, debe reiniciarse con:

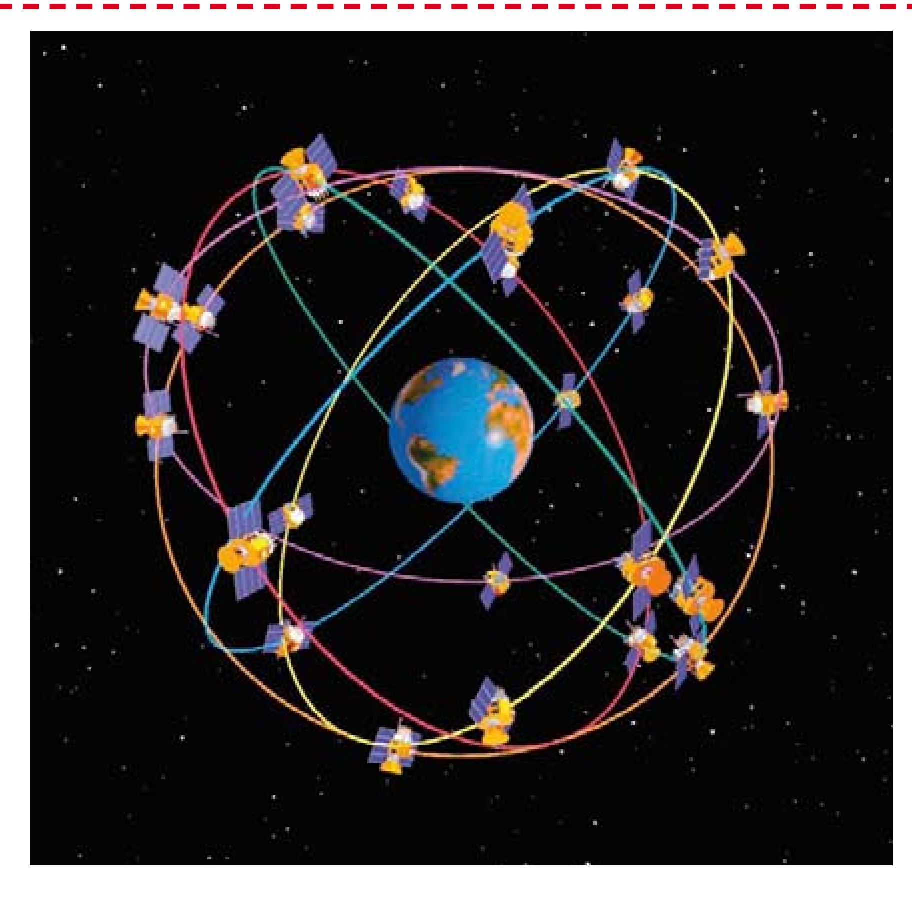
```
at+cfun=1
```

Al cabo de un momentito, si todo ha ido bien, parpadeará el LED superior (D12). Esto significa que el módem interno ha conseguido registrarse en la red. Eventualmente, también puede comprobarse con el comando

```
at+cops?
```

Mejor que GPS

El Global Positioning System, o GPS, estaba destinado al principio a aplicaciones militares. Pero cada vez más se exigía una localización más precisa y poco a poco el sistema GPS se fue abriendo al consumidor. Primero, los americanos, los que diseñaron el sistema, hicieron deliberadamente la señal de satélite más imprecisa. Hace unos años se eliminó esta limitación y ahora el consumidor puede hacer uso de la resolución completa del sistema. Con ello, se puede determinar una posición con el GPS a tres metros de precisión. El nivel de resolución depende del número de satélites "visibles" y si se recibe una señal WAAS o EGNOS.



Los términos WAAS y EGNOS se refieren a sistemas que hacen posible una localización más precisa junto con el GPS. WAAS significa Wide Area Augmentation Sistema y EGNOS European Geostationary Navigation Overlay Service. Lo que el sistema EGNOS hace en Europa, lo hace el sistema WAAS en Norteamérica. Se trata de una red de satélites y estaciones terrestres que genera una señal de corrección de GPS, que da una localización más precisa; de media unas cinco veces mejor. Un receptor que soporta WAAS/EGNOS da en más del 95% de los casos una posición que difiere menos de 3 metros.



EGNOS consta por el momento de tres satélites geoestacionarios y una serie de estaciones terrestres, distribuidas por Europa. Estas estaciones recaban la información de todas las demás estaciones terrestres y generan una señal de corrección. Las estaciones terrestres saben su propia localización a la perfección y la calculan con la señal que reciben de los satélites. Además, esta señal de satélite incluso puede pasar, por ejemplo, a través de condiciones atmosféricas. Entonces, la señal de corrección se transmite a los satélites geoestacionarios. La información coincide con la señal de GPS estándar, lo que significa que cualquier receptor GPS que soporte WAAS puede leer esta señal.

EGNOS es un proyecto conjunto de la AEE, Eurocontrol y la Comisión Europea, y funciona tanto para el sistema de navegación americano GPS como para el Glonass ruso. El número de satélite EGNOS para Europa es el 33. Generalmente, los sistemas GPS indican cuando reciben una señal de corrección. Para ello también se usa el término "differential".

Por otra parte, en Asia se hace uso del sistema MSAS, que funciona del mismo modo.

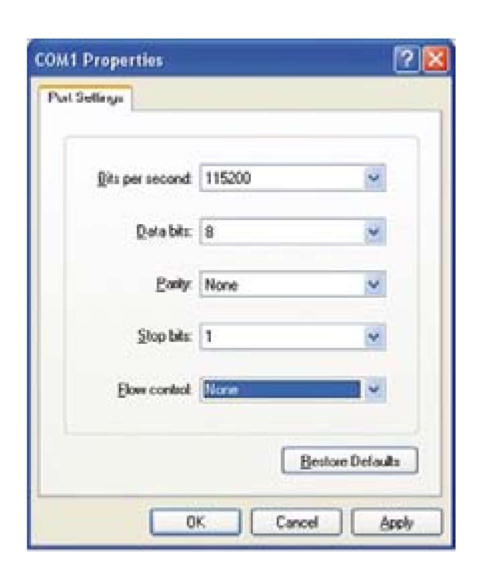


Figura 2. Estos son los parametros adecuados para poder establecer la comunicación serie.

Entonces, el módem responde con
'+COPS: 0,2,20408
OK'
(20408 es el número de operador y es distinto en cada proveedor).
Ahora el módem se registrará automáticamente en la red en cuanto el módulo tenga tensión.

Seguridad

En primer lugar debe ajustarse la contraseña. Por defecto es "elektor". El ajuste tiene que hacerse de la siguiente manera. Debe preguntarse la contraseña con:
`at+password?`

El módem responde con '+PASSWORD: elektor'. Para cambiarla, se usa el comando
`at+password="gpsmodule"`

Atención: las comillas también deben introducirse, de lo contrario se obtendrá un mensaje de ERROR. La contraseña puede tener como máximo 20 caracteres.

Estado del GPS

El módulo GPS se conecta al puerto de serie del módem. En la primera versión de firmware, se han introducido dos comandos con los que se pueden sacar datos del módulo GPS. El primer comando es:
`at+gpshealth?`

Una respuesta del módem será, por ejemplo:

```
'Rcvr status code = 0x01
(Don't have GPS time yet)
Receiver health byte = 0x11
Battery backup: BBRAM not
available at start-up
Antenne feedline fault: An-
tenna line open/short
Type of fault: Open detected
OK'
```

De ahí se puede concluir que el módulo GPS todavía no tiene coordenadas válidas. El estado de la antena muestra que se encuentra en una situación de error. El tipo de error indica que el circuito de la antena tiene una conexión abierta. Probablemente la antena no esté conectada.
Si la antena está conectada y se sigue pidiendo el comando, entonces se recibe la siguiente respuesta:

```
'Rcvr status code = 0x00 (Do-
ing position fixes)
Receiver health byte = 0x01
Battery backup: BBRAM not
available at start-up
Antenne feedline fault: OK
OK'
```

Ahora vemos que el estado de la antena es "OK". Cuando la entrada de la antena se pone en cortocircuito, entonces aparecerá el mensaje 'Short Circuit'. El código de estado del receptor da el mensaje de que ha podido determinar una posición válida. Cuando el módulo ya está activado, el código de estado del receptor indica el número de satélites recibidos. En cuanto el módulo haya podido determinar una posición válida, el LED inferior (D3) parpadeará. Si el LED es continuo, entonces significa que algo no va bien. Puede ser que el módulo reciba demasiado pocos satélites, pero también puede ser que la antena esté mal conectada.

En uso

Hemos constatado que hay una conexión y que queremos saber las coordenadas de longitud y latitud. Para ello, damos el comando
`at+gpsposition?`



Figura 3. La nueva localización de nuestra oficina no supone ningún problema para el ElekTrack

Seguidamente, el módem nos devuelve sus coordenadas, por ejemplo

```
Long: 5.803043 E;
Lat: 50.941492 N;
OK
```

Ahora, naturalmente, también queremos poder pedirlo por SMS, de modo que podamos pedir la localización del módulo a distancia. Para evitar que todos reciban sin más una respuesta del módulo cuando se le envía un SMS éste debe estar provisto de una contraseña. Para ello enviamos el siguiente SMS al módulo:

```
info:<password>:<phonenum>
```

O bien

```
INFO:<password>:<phonenum>
```

Atención: "info" debe escribirse todo en mayúscula o todo en minúscula. En el lugar de <password> debe insertarse la contraseña configurada con el comando at+password.

El término <phoneum> es opcional. Por defecto la respuesta se envía al remitente. Si se desea enviar la respuesta a otro número, entonces se puede hacer con la ayuda de este parámetro.

Tras unos segundos, el ElekTrack enviará un SMS con los datos de longitud, latitud y altitud, con lo cual podemos determinar la ubicación del ElekTrack. Esto se puede hacer por ejemplo en línea en [3] (**figura 3**) y [4].

El ElekTrack está disponible y se puede encargar como módulo totalmente montado en nuestra webshop en www.elektor.es. La última versión de firmware también se encuentra en nuestro sitio. Y, por supuesto, siempre estamos encantados de conocer las ideas y aplicaciones que se inventan nuestros lectores.

(040161)

Vínculos:

[1] http://www.carmobile.nl/autoalarm.htm

[2] http://nl.wikipedia.org/wiki/Galileo_(navigatie)

[3] www.gpscoordinates.eu

[4] http://boulter.com/gps

Problemas prácticos

Todo aquél que alguna vez haya escrito software, puede hablar de los pequeños gusanos que siempre se cuelan en el código. En la fase inicial de la escritura del software tuvimos problemas con el envío de SMS. Parecía que el servicio de SMS del módem no quería funcionar. Tras horas de búsqueda y lectura de documentación, se nos ocurrió la simple y brillante idea de probar con otra tarjeta SIM. Así que nos fuimos inmediatamente a la tienda a buscar una nueva tarjeta SIM. ¿Y cuál fue nuestra sorpresa?: con una nueva tarjeta ya no tuvimos más problemas.

Sin embargo, nos sorprendió mucho que sí que funcionara con una nueva tarjeta SIM. Así que nos pusimos en contacto con el proveedor. Efectivamente, resultó que había un gusano en el firmware del módem. Por lo visto, en algunas tarjetas SIM el registro del servicio SMS no funcionaba bien. En la nueva versión de firmware, este problema ya se ha solucionado.

Otro problema que seguro que conocen los diseñadores de software es la diferencia little-endian y big-endian en la configuración de memoria. Las coordenadas de longitud y latitud se muestran como radianes en forma de dobles del módulo GPS. El módulo funciona en el modo de byte little-endian, mientras que el módulo GSM en el modo byte big-endian. En la introducción de los datos (como valores ASCII) el orden de los bytes debe invertirse. Y es que cuando se convierten datos little-endian en un procesador big-endian en forma de radianes, el resultado no cuadra con ninguna medición. Al principio no nos dimos cuenta. Pues lo primero que piensas es que has cometido un error en el código que debe hacer la conversión.

Más tarde parece una tontería, pero en ese momento te tiras de los pelos.



PLC ECIO

Barato, casero y con CAN!

Por Ben Rowland (Matrix Multimedia) & Luc Lemmens (Elektor Labs)

Aquí tenemos la primera aplicación real del los módulos ECIO introducidos en el número de Octubre de 2007 en Elektor. Un ECIO actúa como el cerebro de una placa PLC que dispone de relés, opto-acopladores, conexión CAN (!) y una pantalla LCD. Toda esta capacidad de E/S, junto con la aplicación Flowcode, pueda funcionar como una PLC versátil y potente para proyectos de automatización y control bastante complejos. El modulo LCD se utiliza para mostrar los caracteres ASCII al usuario, como medio de reparación durante la fase de desarrollo de los programas o para monitorizar el sistema.

Prestaciones de la PLC ECIO

- Conexión a bus CAN
- 4 entradas aisladas ópticamente
- 4 salidas controladas por relé
- Pantalla LCD alfanumérica de 2x16 caracteres
- Programada con Flowcode, pero acepta ficheros hex para PIC18
- Flowcode gratuito para ECIO
- Conexión USB para programación

Una placa de aplicación PLC completa, que incluya salidas aisladas, entradas, pantalla LCD y conexión con el bus CAN… suena muy bien! Pero también nos gustaría oír que se trata de una placa de bajo coste, basada en un módulo ECIO y fácil de programar utilizando la aplicación Flowcode, un método totalmente gráfico para programar PICs (y también otros microcontroladores).

PLC

Un PLC (Programmable Logic Control, es decir, "Control Lógico Programable") es un dispositivo utilizado normalmente como elemento de control central inteligente, en un proceso industrial diseñado con diagramas de flujo, habitualmente para fabricación en masa o verificación de calidad. Los controles de cintas transportadoras se hacen tradicionalmente por medio de PLCs.
El PLC y sus lenguajes de programación (de alto nivel) se han estandarizado y se han extendido ampliamente. Por desgracia, la factura de casi cualquier PLC comercial lo aleja rápidamente del alcance de cualquier aficionado a la electrónica. Una pena, ya que muchos lectores de Elektor tienen bastante relación con los sistemas de control industriales.
Una secuencia de comandos PLC es similar a los programas de un ordenador: siguen una secuencia predefinida de eventos con todas las opciones para bucles condicionales, temporizadores de eventos, valores analógicos (temperatura, nivel de líquidos, presión) y un sencillo registro de resultados.
Si tenemos un proceso relativamente más complejo es posible que queramos realizar ese control utilizando electrónica sobre la máquina. En ese caso, una placa PLC es una buena opción debido a su gran robustez eléctrica y a su flexibilidad, cuando se trata de conectar el mundo real a través de relés y opto-acopladores.
Para ser capaces de programar un PLC (por ejemplo, definir la secuencia de eventos que suceden en un proceso) necesitamos escribir un programa. Una vez depurado y simulado sobre un ordenador, podemos volcar el programa sobre el PLC y, a partir de aquí, "cruzaremos los dedos".

Los programas de un PLC pueden ser editados, depurados, ampliados, optimizados y, por supuesto, almacenados y recuperados. Justo igual que sucede con un programa de un microcontrolador o de un ordenador!

Descripción del circuito

En la **Figura 1** se muestra el esquema eléctrico de la placa PLC ECIO. El circuito comprende un cierto número de componentes variados que vamos a tratar a continuación.

ECIO

El módulo ECIO es el cerebro de la placa. Es utilizado para controlar todos los dispositivos periféricos que hay sobre la placa. Los módulos ECIO, introducidos en el número de

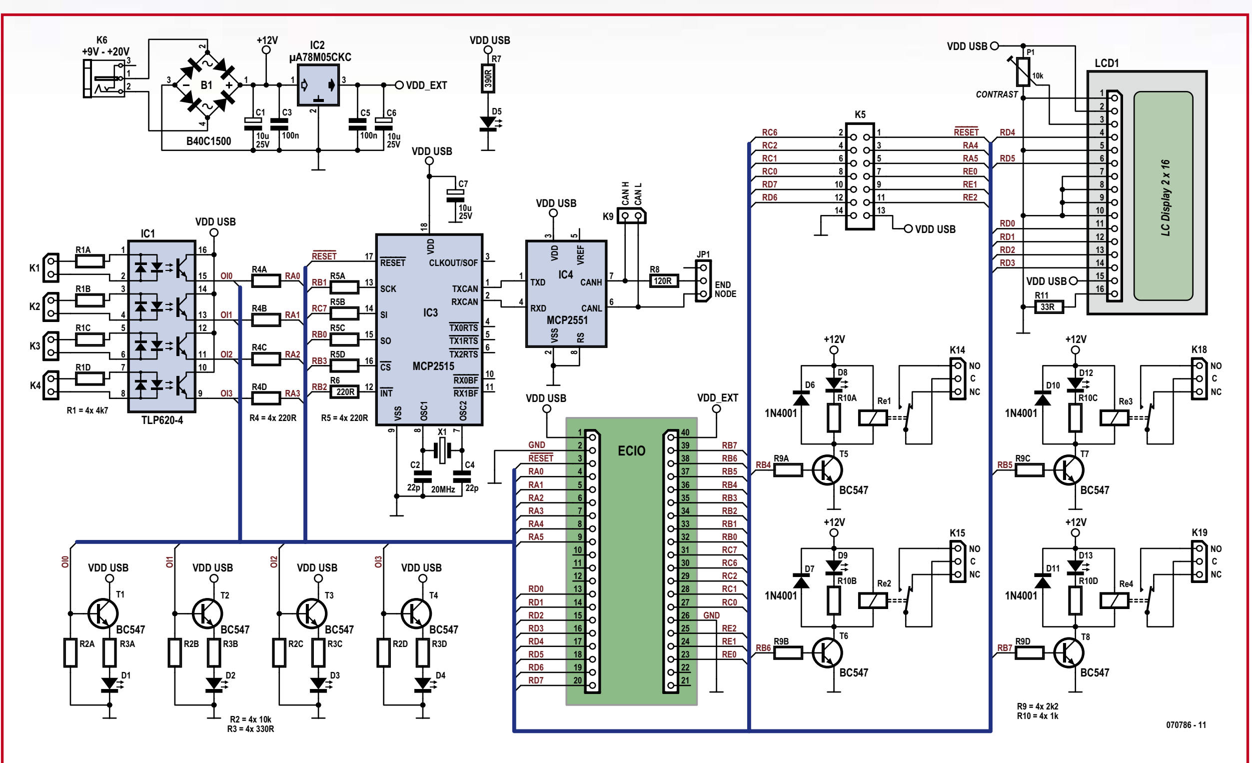


Figura 1. Esquema eléctrico del circuito de la placa PLC ECIO. Componentes mínimos – gran potencial en términos de E/S.

octubre de 2007 de *Elektor* [1], representan un método de muy bajo coste de entrar en el mundo de la programación de microcontroladores PIC. Los módulos ECIO están disponibles en la tienda de Elektor, con muy buenos descuentos para pedidos elevados. En este caso hemos utilizado la versión de 40 terminales denominada ECIO-40P.

Relés

Los relés de la PLC se utilizan para conmutar dispositivos electrónicos (activar/desactivar) y son controlados a través de los terminales RB4–RB7 de la ECIO. La programación de un nivel lógico bajo en estos terminales desconecta el relé correspondiente, mientras que su puesta a nivel lógico alto

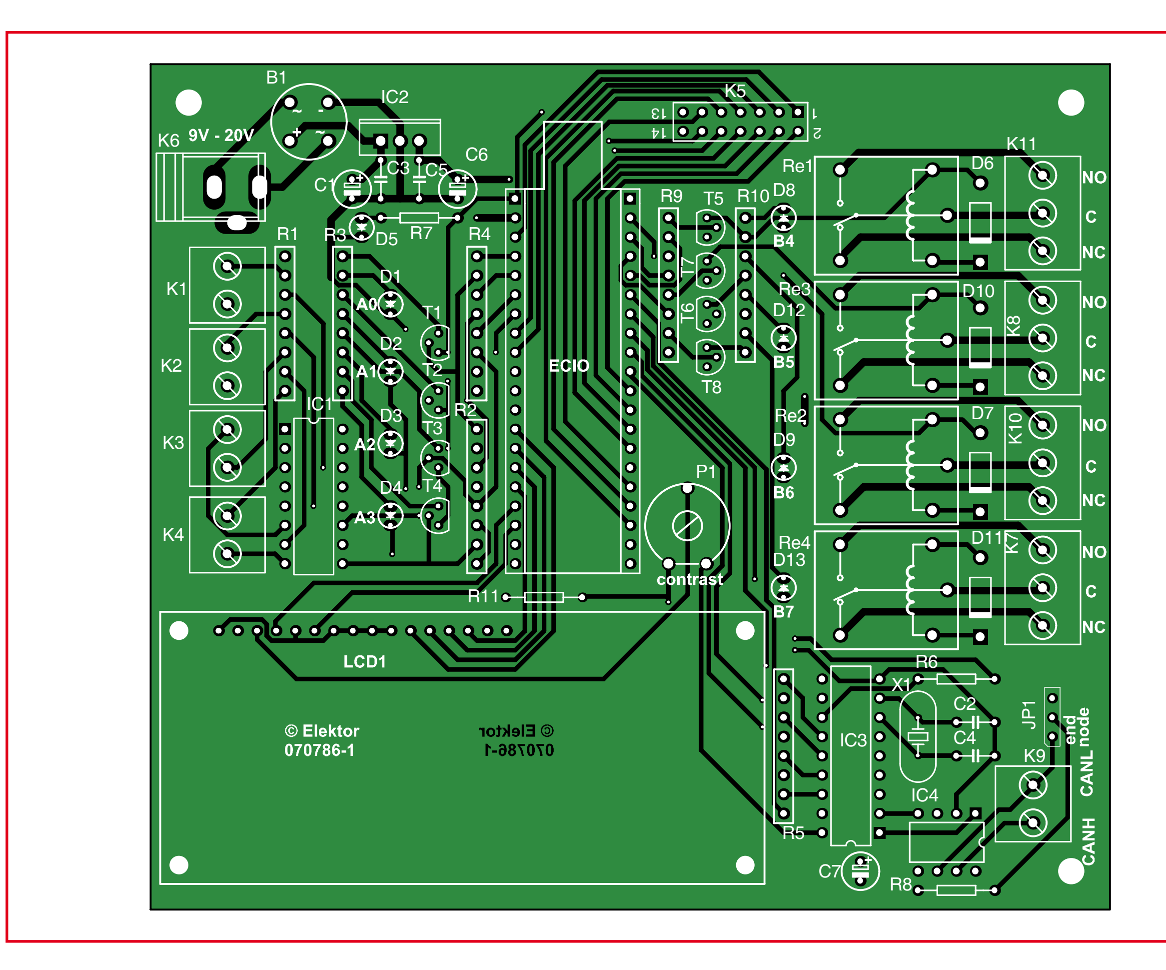


Figura 2. La placa PLC ECIO sin componentes está disponible en Elektor.

Lista de materiales

Resistencias
R1 = 8-pin SIL array 4 x 4kΩ7*
R2 = 8-pin SIL array 4 x 10kΩ*
R3 = 8-pin SIL array 4 x 330Ω*
R4,R5 = 8-pin SIL array 4 x 220Ω*
R6 = 220Ω
R7 = 390Ω
R8 = 120Ω
R9 = 8-pin SIL array 4 x 2kΩ2*
R10 = 8-pin SIL array 4 x 1kΩ*
R11 = 33Ω
P1 = 10kΩ preset
* ver texto

Condensadores
C1,C6,C7 = 10µF 25V radial
C2,C4 = 22pF
C3,C5 = 100nF

Semiconductores
D1-D5,D8,D9,D12,D13 = 3mm LED
D6,D7,D10,D11 = 1N4001
B1 = B80C1500 (encapsulado redondo; 80Vpiv @ 1.5Ap)
T1-T8 = BC547
IC1 = TLP620-4
IC2 = 7805
IC3 = MCP2515-I/P
IC4 = MCP2551-I/P

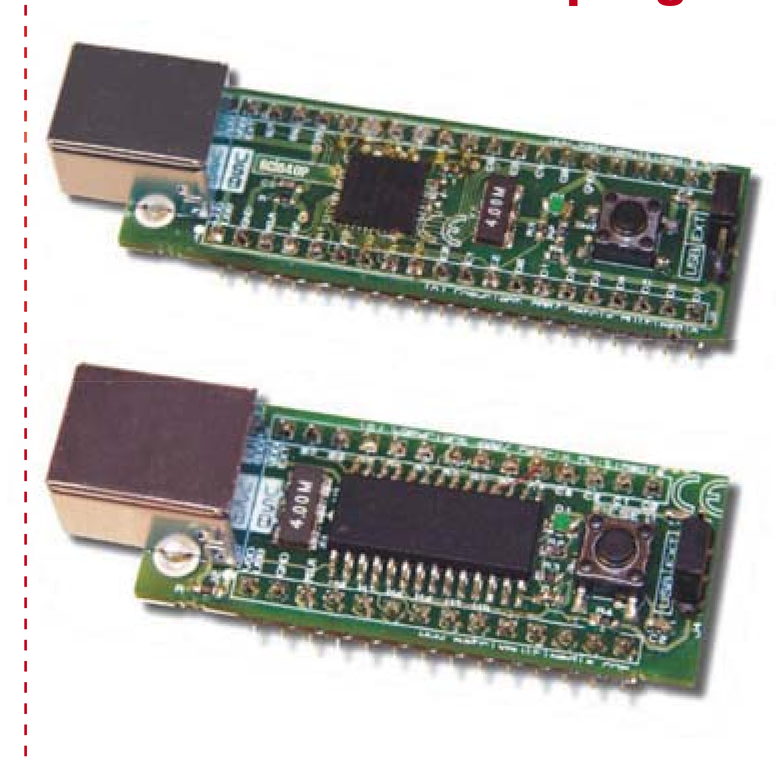
Varios
Re1-Re4 = 12V relé, SPDT, ejemplo, Omron G5LE-1.
X1 = cristal de cuarzo de 20 MHz.
K1,K4,K9 = Bloque terminal de 2 contactos con tornillos, separación entre terminales de 5mm.
K5 = Conector "boxheader" de 14 terminales.
K6 = Conector de tensión de red AC/DC de baja tensión, para montaje en circuito impreso, ejemplo, CUI Inc. # PJ-002B (Digikey # CP-002B-ND) o Cliff Electronic Components # DC10B (Farnell # 224960).
K7,K8,K10,K11 = Bloque terminal de 3 contactos con tornillos, separación entre terminales de 5mm.
ECIO = ECIO-40P módulo procesador (Elektor Shop).
LCD1 = pantalla LCD, alfanumérica, 2x16 caracteres, ejemplo, Displaytech 162.
JP1 = Conector "pinheader" de 3 terminales SIL con Puente.
Placa de circuito impreso, código de pedido 070786-1 en la tienda de Elektor.

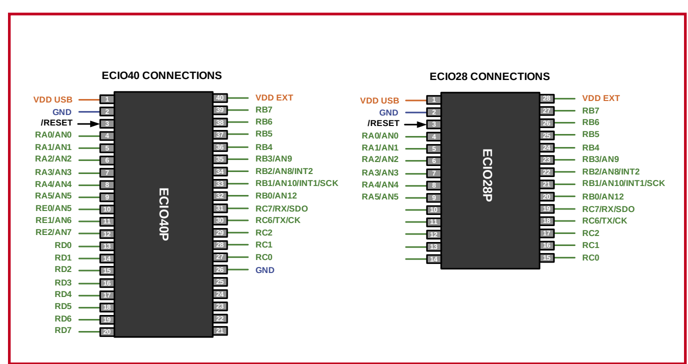
implica la activación de la bobina del relé asociado. Si un relé está activado, el diodo LED situado al lado de ese relé concreto se encenderá para dar una representación visual de encendido/apagado. Los relés son utilizados para proporcionar un aislamiento eléctrico entre el modulo ECIO y las tensiones de conmutación externas (las cuales pueden llegar a ser de 230 V de tensión de red). Esto significa que los dispositivos de potencia que trabajan con tensiones elevadas como 48 VDC o 230VAC, pueden ser controlados de manera segura a través del módulo ECIO.
Cada conjunto de contactos de relé (NO y NC) sale de la placa a través de un bloque de terminales atornillados para montaje en circuito impreso, de tres salidas (el terminal central (C) es el que hace la conmutación).

Opto-aisladores

Las entradas opto-aisladas son especiales en el sentido de que no disponen de terminales más (+) y menos (–). Dentro de IC1, los dos diodos de cada entrada son, de hecho, diodos LED, de manera que la polaridad de la tensión de control que deseamos aplicar a la placa PLC ECIO no nos afecta. Las salidas de los opto-aisladores es-

ECIO — el PIC USB programable más barato





La familia ECIO de microcontroladores programables por USB proporciona una manera sencilla de adoptar la tecnología de microcontroladores en nuestros proyectos. El componente se comporta como un microcontrolador normal, pero cuando conectamos los terminales USB y usamos el conmutador de "reinicio" ("reset"), podemos enviar un nuevo programa al componente para su programación. Esto hace que el ECIO sea uno de los PICs programables, compatibles USB, de más bajo coste en el mundo.

Actualmente solo hay dos componentes en la familia: el ECIO-28P y el ECIO-40P. Estos dispositivos están basados en las series PICmicro 18; en concreto, el 18F2455 y el 18F4455, respectivamente.

Los microcontroladores ECIO están preprogramados con un programa "cargador de arranque" que nos permite enviar un nuevo programa al microcontrolador a través del puerto USB, en principio, tantas veces como deseemos. ECIO es compatible con ficheros en código hexadecimal provenientes de cualquier compilador apropiado. ECIO también es compatible directamente con la aplicación Flowcode (un lenguaje de programación gráfico que nos simplifica enormemente el proceso de generación de código), pero que también es utilizado con cualquier compilador C o con entornos de desarrollo propios a la casa Microchip como MPLAB.

ECIO se suministra también con un amplio abanico de herramientas de desarrollo y de aprendizaje, incluyendo Flowcode (gratuito) y Bloques Electrónicos muy baratos.

tán conectadas a los terminales RA0 a RA3 del modulo ECIO. Estos terminales de entrada del módulo estarán a nivel lógico cero cuando no tengamos tensión de entrada y a nivel lógico uno para tensiones de 3,5 V o superiores. Utilizamos los opto-aisladores para proporcionar una capa de aislamiento de las tensiones de entrada. Esto significa que las tensiones relativamente elevadas pueden ser utilizadas de forma segura para controlar el módulo ECIO. Las señales que queremos procesar en la placa PLC ECIO son aplicadas al circuito a través de los bloques de terminales atornillados para montaje en PCI de dos líneas. Los diodos LEDs D1 a D4 muestran el estado lógico de las entradas opto-aisladas.

CAN

El interfaz CAN (Controller Area Network, es decir, Controlador de Red de Área) se utiliza para añadir la placa PLC ECIO sobre una red CAN. El bus CAN está de moda y conocemos muy pocos PLCs que ofrezcan esta conexión. En nuestro montaje usamos un circuito integrado controlador CAN MCP2515 y un controlador de línea MCP2551. El circuito integrado controlador CAN está conectado a los terminales periféricos del modulo ECIO, configurados para establecer una comunicación SPI. También dispone de un terminal de interrupción que está conectado al módulo ECIO a través del terminal RB2 y de un terminal de selección de circuito ("chip select"), el cual se conecta al módulo ECIO por medio del terminal RB3. El controlador CAN tiene su propio reloj de 20 MHz, obtenidos del cristal de cuarzo X1. La conexión actual del bus CAN al modulo ECIO se realiza a través del conector K9, un bloque de terminales atornillados para montaje en circuito impreso de dos líneas. El puente JP1 tiene que ser instalado tan solo si el controlador de línea MCP2551 está en el nodo final del bus.

LCD

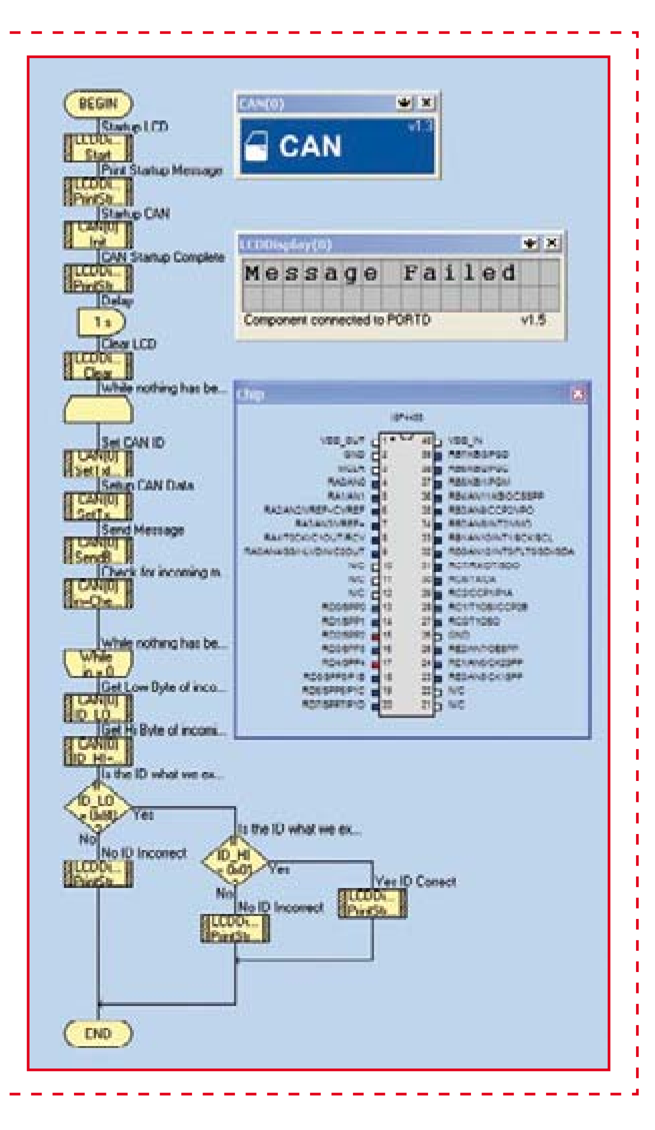
El módulo LCD es utilizado para mostrar caracteres ASCII al usuario como medio de apoyo en la fase de desarrollo de los programas, o para monitorizar el funcionamiento del sistema cuando está ocupado verificando y controlando los eventos en un proceso automatizado. La pantalla LCD está conectada al modulo ECIO por medio de los terminales RD0 a RD5, con los 4 bits de datos ocupando los bits

Verificación del interfaz CAN

El bus CAN es especialmente bueno para enviar estructuras complejas de mensajes entre un cierto número de ECUs (microcontroladores). El programa de prueba escrito en la aplicación Flowcode busca sencillamente un "eco" del mensaje CAN saliente. Esto podría ser utilizado para ver cuántos nodos están en la red o utilizarlo para calcular la distancia entre nodos, basándonos en el tiempo del eco, etc.

Básicamente, el programa de prueba para la placa ECIO está enviando un mensaje CAN específico con un ID estándar de 12. Cuando el Multi-programador recibe un mensaje CAN, verifica el ID y, si es igual a 12, devuelve el ID 12 del mensaje. La placa ECIO continúa enviando este ID de mensaje hasta que recibe un mensaje CAN de vuelta. Una vez que ha recibido un mensaje de vuelta, verifica el ID del mensaje y confirma o deniega si el "eco" era correcto.

Para esta prueba necesitaremos otra placa PLC ECIO o un Multi-programador de e-Blocks conectado a un módulo CAN de e-Blocks. Una placa PLC ECIO enviará varios mensajes sobre el bus CAN. La otra, permanecerá a la escucha hasta que aparezca un mensaje predefinido y responda así hacia la placa ECIO origen. Cuando el transmisor CAN es iniciado, la pantalla presenta el mensaje 'Startup' ("Arranque"), seguido algo más tarde por el mensaje 'Done' ("Hecho"), una vez que el controlador CAN ha sido inicializado. Si la conexión CAN es correcta, la pantalla mostrará el mensaje 'Message returned' ("Mensaje devuelto") si no, veremos 'Message failed' ("Mensaje fallido").



0–3, el bit RS ocupando la posición del bit 4 y el bit Enable tomando la posición del bit 5. P1 es el potenciómetro de ajuste del contraste de la pantalla LCD.

Fuente de alimentación

No hay nada de especial en este apartado: el tradicional regulador de tensión 7805 (IC2x) y el habitual conjunto de condensadores de desacoplo. Delante del regulador se utiliza un puente rectificador para permitir que la placa PLC ECIO pueda ser alimentada, tanto por tensión AC como DC, teniendo en cuenta que la tensión de entrada a la placa debe estar comprendida entre 9 V y 20 V. Para una alimentación con VDC la polaridad es irrelevante.

Bueno, como la placa PLC ECIO tiene una gran cantidad de conexiones hacia el mundo exterior y para evitar el perderse siguiendo todas estas líneas de E/S y sus componentes asociados, en la **Tabla 1** hemos dado un resumen de todo ello.

Montaje

En la **Figura 2** se muestra el plano de montaje de componentes del circuito impreso diseñado para la placa PLC ECIO. Como es habitual, los planos de las pistas de cobre pueden ser descargados de nuestra página web, para todos aquellos que dispongan de las herramientas adecuadas para tratar con ácido y taladrar las placas de circuito impreso en casa. Para el resto, suponemos que les agra-

Tabla 1. Resumen de las conexiones del ECIO

Relé de salida	Terminal E/S del ECIO
Relé Re1 (K14)	RB4
Relé Re2 (K15)	RB5
Relé Re3 (K18)	RB6
Relé Re3 (K19)	RB7
Entrada opto-aisladas	**Terminal E/S/LED del ECIO**
K1	RA0 / D1
K2	RA1 / D2
K3	RA2 / D3
K4	RA3 / D4
CAN (desde la CPU)	**Terminal E/S del ECIO**
Salida datos serie	RC7
Entrada datos serie	RB0
Reloj serie	RB1
Interrupción	RB2
Chip Select	RB3
LCD	**Terminal E/S del ECIO**
D0	RD0
D1	RD1
D2	RD2
D3	RD3
RS	RD4
Habilitar	RD5

dará oír que la placa, sin componentes pero ya fabricada, está disponible en la tienda de Elektor.
En este proyecto no se han utilizado componentes SMD, por lo que el montaje de los distintos elementos sobre la placa es bastante sencillo, con tan solo poner algo de cuidado y precisión en la soldadura de los componentes. El grupo de resistencias de ocho terminales, que contiene cuatro resistencias **cableadas de forma individual** (es decir, no tienen un punto común), pueden ser algo difíciles de conseguir. En lugar de este grupo de resistencias, Podemos utilizar cuatro resistencias individuales montadas de forma vertical.
Una vez que tenemos la placa completada, haremos una inspección visual detenida antes de insertar los integrados y alimentarla por primera vez.

Probando el circuito ECIO - E/S y CAN

Pará ayudar a nuestros lectores a localizar los fallos, hemos escrito un sencillo e instructivo programa de prueba ECIO, cuyo listado de Flowcode se muestra en la **Figura 3**. Así pues, lo descargaremos, lo volcaremos sobre la aplicación Flowcode, ejecutaremos la simulación, lo compilaremos y lo pasaremos sobre el módulo ECIO.
El programa incluye una rutina de configuración general del PIC que hace toda la labor de configurar todos los puertos y los terminales de E/S del módulo ECIO. Por ejemplo, las líneas RB4 – RB7 del modulo ECIO son configuradas como líneas de salida (para controlar los relés). El programa de prueba es bastante sencillo de usar: si activamos el opto-aislador de la línea RA0, el relé sobre la línea RB4 se activará. Algo similar pasa con la combinación de las líneas RA1/RB5, y con el resto. La pantalla LCD nos muestra el mensaje: 'Elektor ECIO PLC board' (¿no es asi?, ¡intentemos ajustar el contraste del LCD!). Ni los miembros de la familia ni este PIC pueden llegar a hacer cosas impresionantes pero, si funcionan adecuadamente, estará ejecutando un completo bloque de código PIC, con lo que podremos asumir de manera segura de todo el circuito está funcionando como debe.
La utilidad de prueba del bus CAN, tratada en un apartado independiente, también está incluida en el bloque de descargas gratuitas para este proyecto. Los dos programas de prueba proporcionados son bastante educativos y verdaderamente valiosos, incluso si no acometemos el montaje de este proyecto (consejo: utilizar la versión gratuita de Flowcode para empezar). El nombre del fichero a descargar es **070786-11.zip**.

Programación de la PLC

Toda la programación de la placa PLC ECIO presenta una gran similitud con los e-Blocks y con Flowcode, por lo que si ya tenemos alguna experiencia con algunos de estos dos elementos, estamos de suerte. Si no, existe una cantidad inmensa de información sobre estos temas en Internet [1], en el número anterior de Elektor y en el propio paquete que acompaña Flowcode. Lo mejor de todo es que Flowcode para ECIO es **gratuito** [1].
La escritura de un programa de control de procesos para la placa PLC ECIO nos confirma que Flowcode lo es todo: en lugar de preocuparnos por la sintaxis y del código ensamblador del PIC con el que estamos trabajando a un nivel bastante elevado, basta con dibujar un diagrama de flujo del programa y dejar que Flowcode se las arregle con todo el proceso de compilación, inicialización, localización de errores, y volcado del código sobre el módulo ECIO. Por supuesto, podemos simular nuestro programa PLC, de forma que tengamos bastantes oportunidades de éxito cuando nuestra placa PLC ECIO sea conectada al mundo real.

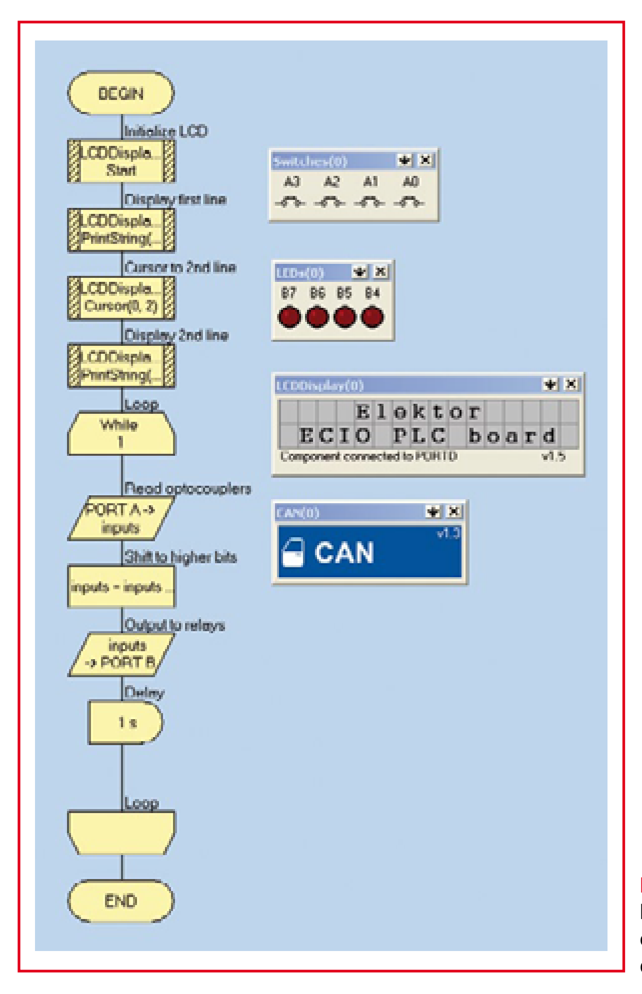


Figura 3. Este programa Flowcode ejecuta un test interno del circuito en la PLC ECIO.

Los usuarios más avanzados es posible que quieran contar con sus propios métodos de generación de código para el PIC18, usando compiladores C++ o similares. El módulo ECIO acepta incluso código hexadecimal directo proveniente de cualquier código ensamblador realizado con otros sistemas. Sencillamente utilizaremos los controladores USB de E/S gratuitos para conectar el módulo ECIO a nuestro PC.

(070786-I)

Enlaces en Internet y Referencias

[1] http://www.matrixmultimedia.com/ECIO-X.php

[2] Control Sencillo de E/S, *Elektor* Octubre 2007.

Adaptador JTAG Universal

Marcel Cremmel



Para programación y emulación

Este adaptador fue pensado originalmente para permitir la programación de la memoria y de la CPLD del microcontrolador PSD 813 utilizado en el cartucho GBECG de la consola Gameboy, el cual convierte esta consola de juegos en un electrocardioscopio (veáse la entrega de Diciembre de 2006). Pero esta interfaz es mucho más universal (ver apartado titulado "JTAG 'In-circuit' – Algunas aplicaciones"). Nuestro adaptador se conecta a un puerto paralelo de un ordenador y utiliza el protocolo JTAG IEEE 1149.1

Por supuesto, los aficionados a la microelectrónica bien informados conocen perfectamente que otros dispositivos programables en el propio circuito ("in-circuit") utilizan este mismo puerto (paralelo) y un protocolo idéntico. Por desgracia, los programadores/emuladores pensados para estos dispositivos no son compatibles (de hecho, son bastante diferentes, de manera que no es posible pensar en una solución intermedia).
Sin embargo, un examen más detenido de los esquemas eléctricos de los circuitos de ciertos programadores, sugeridos por varios fabricantes de circuitos integrados, muestran que las diferencias son relativamente menores y que, de hecho, dichas diferencias se basan principalmente en la interconexión entre las señales del puerto LPT y las del conector JTAG. Por lo tanto, lo único que tenemos que hacer es crear unas cuantas funciones de multiplexado para obtener un adaptador "universal".
Estos dispositivos pueden conseguirse perfectamente utilizando componentes lógicos convencionales. Así, el circuito de nuestro adaptador podría haber sido bastante más complejo, con diferente electrónica para cada una de las secciones de los diferentes tipos de procesador. Utilizando un circuito lógico programable EP 900 (de la casa Altera, que se ofrece gratuitamente desde Elektor), es posible ofrecer un programador muy sencillo y barato.
Muchos fabricantes han adoptado el pro-

tocolo JTAG (Join Test Action Group, es decir, Grupo de Acción de la Asociación de Pruebas) para la programación, depuración y comprobación de sus circuitos integrados sobre la propia placa de trabajo (circuitos integrados para trabajar "en el propio circuito"). Afortunadamente, no necesitamos saber todos los detalles de este protocolo para utilizarlo: el programa que corre sobre el ordenador (normalmente gratuito) y los componentes de destino, incluyen cada uno un núcleo JTAG que les permite comunicarse entre ellos de un modo completamente transparente.

Los dispositivos relacionados con este proyecto tienen unos terminales "JTAG" especiales, de manera que lo único que tenemos que hacer es conectar dichos terminales a los del mismo nombre que aparece en el conector del programador. El tamaño (número de contactos) y la distribución de terminales de este conector, difiere de un fabricante a otro. Esta información la proporcionamos a nuestros lectores a través de los distintos esquemas eléctricos que se muestran a lo largo de las Figuras 1- 4, las cuales se corresponden respectivamente con los dispositivos CPLDs y EPLDs (Byteblaster II) de la casa Altera (**Figura 1**); CPLDs y EPLDs (Cable de Descarga Paralelo) de la casa Xilinx (**Figura 2**); microcontroladores MSP 430 de la casa Texas Instruments (Comunicación JTAG con LPT IF de 4 hilos) (**Figura 3**); y las familias PSD, μPSD y DSM (Flashlink FL 101) de la casa ST Microelectronics (**Figura 4**). También debemos hacer notar que existe una cierta discrepancia en el nombre dado a las distintas señales entre los diferentes conectores JTAG.

CIRCUITO ADAPTADOR

El corazón del circuito (vea la **Figura 5**), que con sus 44 terminales difícilmente pasa inadvertido, es una PLD EP 900. Esta PLD forma el enlace entre el puerto paralelo del ordenador, K1, y los cuatro conectores JTAG del tipo DIL "pinheader" para los cuatro dispositivos de destino, nombrados respectivamente como MSP 430 (K2), FLASHLINK (K3), XILINX (K4) y ALTERA (K5). El conmutador SW, un conmutador DIP doble que comprende los contactos de JP1 y JP2, permite la selección de uno de los cuatro tipos de programadores reconocidos por el adaptador JTAG (ver tabla de verdad en el esquema eléctrico del circuito, que también se proporciona sobre la cara de componentes de la placa). Estas cuatro opciones aparecen en la forma de tabla de estado que identifica la cabecera HE-10, en la parte inferior izquierda del circuito. Cada opción tiene su propia estructura lógica dentro de la EP 900. Todos es-

JTAG 'In-Circuit' – Algunas Aplicaciones

- PSDs, uPSDs y DSMs de la casa ST Microlectronics
- Microcontroladores MSP430 de la casa Texas Instruments
- EPLDs y CPLDs de la casa ALTERA
- EPLDs y CPLDs de la casa XILINX

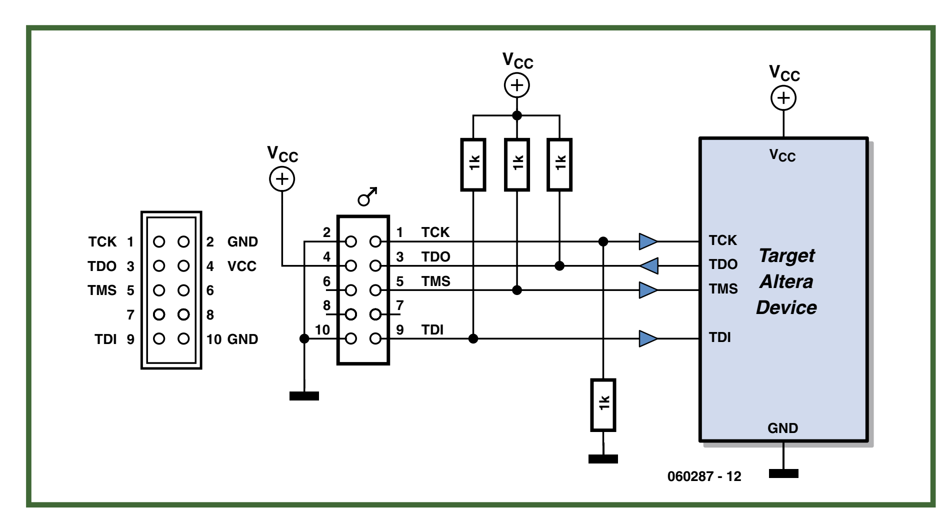


Figura 1. CPLD y EPLD (Byteblaster II) de la casa Altera: conector DIL de 10 terminales. Programas: Quartus II Web Edition, Quartus II Programmer [1].

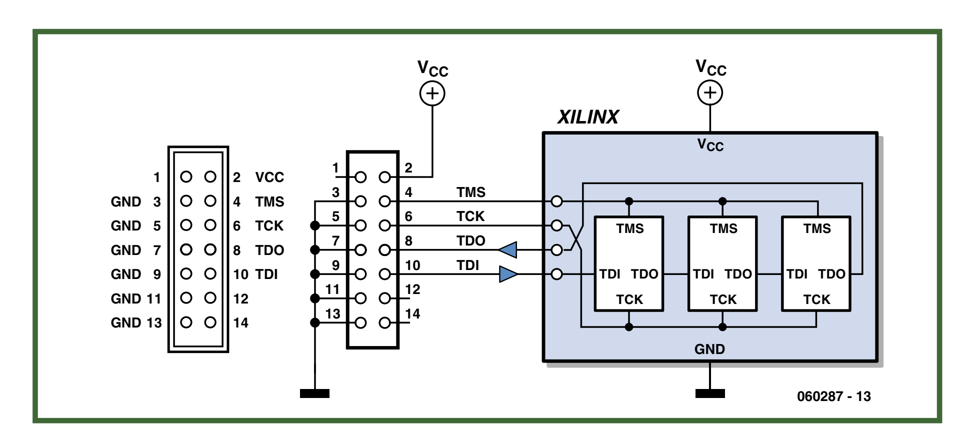


Figura 2. CPLD y EPLD (Cable de Descarga Paralelo) de la casa Xilinx: conector DIL de 14 terminales. Programas: ISE WebPACK [2].

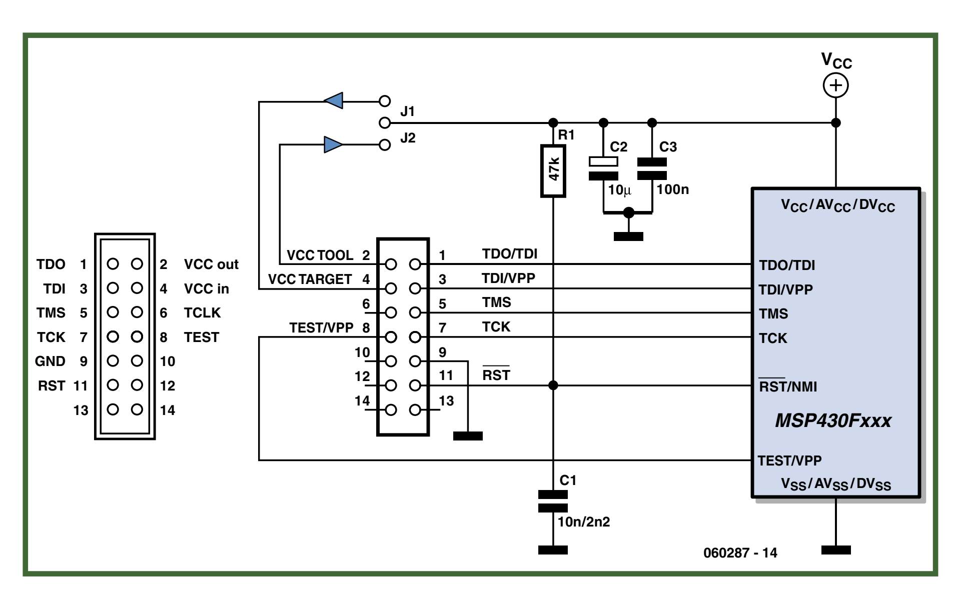


Figura 3. Microcontrolador MSP 430 (Comunicación JTAG - LPT-IF de 4 hilos) de la casa Texas Instruments: conector DIL de 14 terminales. Programas: IAR-Kickstart [3].

Acerca del autor

Marcel Cremmel, el autor de este artículo, tiene el título de ingeniero eléctrico, especialidad en electrónica, desde 1979 (certificado por el Sistema de Educación Nacional Francés).

Después de haber completado su primer año de aprendizaje en la Escuela de Ingeniería de Rabat, en Marruecos, bajo un sistema de cooperación, en 1982 fue asignado al Colegio Louis Couffignal, en Strasbourg, en la sección BTS SE (Certificado de Técnico Superior, "sistemas electrónicos").

Su trabajo requiere que sus conocimientos cubran todos los campos de la electrónica, aunque sus preferencias son las telecomunicaciones, el vídeo, los microcontroladores (MSP 430 y PIC) y los dispositivos lógicos programables (de la casa Altera).

Además de la electrónica, su otra pasión son las motocicletas en todas sus formas: turismo, competición, etc., su página web personal es http://electronique.marcel.free.fr/

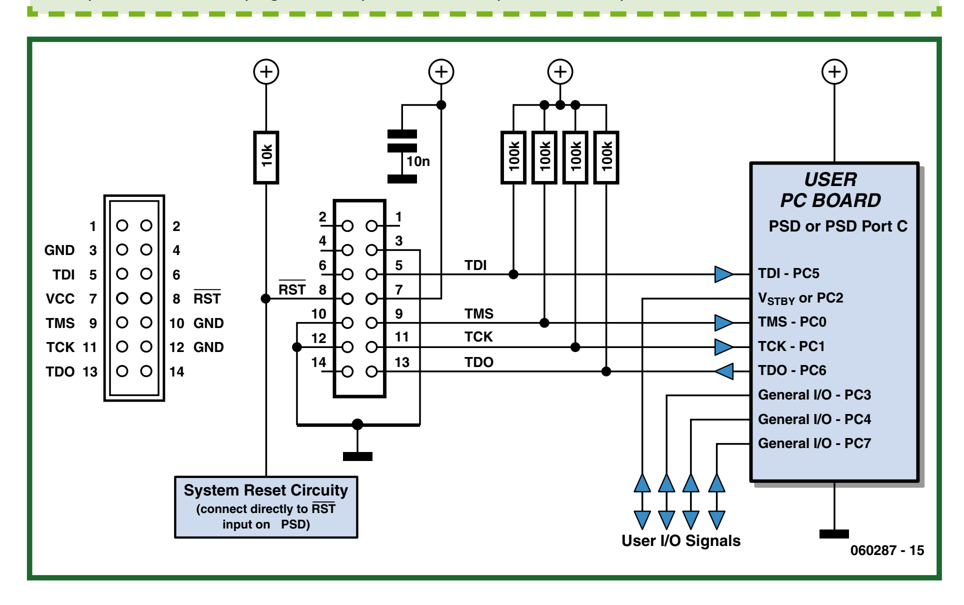


Figura 4. **Familias PSD, μPSD y DSM Flashlink FL-101 de la casa ST Microelectronics: conector DIL de 14 terminales. Programas: entre otros, PSDsoft Express [3] para programar la PSD 813 en el cartucho de la GameBoy.**

tos subconjuntos diferentes utilizan puertas lógicas con diferente configuración, tal y como se muestra en la **Figura 6**.
Cada una de estas estructuras ha sido tomada a partir de los circuitos programadores de los fabricantes. Por razones de eficiencia, la estructura lógica de la EP 900 está descrita en el lenguaje AHDL de la casa Altera. El circuito eléctrico es bastante sencillo de leer para un técnico electrónico, pero en este caso es más eficiente la forma "AHDL". Para información de nuestros lectores, en el apartado correspondiente de este artículo se entregará el listado completo del fichero "fuente" (.tdf) de la EP 900.
En la parte inferior izquierda del esquema eléctrico encontramos la …

FUENTE DE ALIMENTACIÓN

¡La PLD EP 900 es ya un circuito integrado bastante viejo! Requiere una tensión de alimentación de 5 V y, aunque su consumo es bastante elevado, el adaptador de programación puede estar alimentado directamente de las salidas del puerto LPT del ordenador. Para simplificar la implementación y permitirnos trabajar con una tensión de alimentación especialmente dedicada, hemos decidido alimentar el adaptador a partir de las líneas de alimentación del sistema de destino. Pero esta tensión de alimentación normalmente está contenida dentro de los valores de 3 a 3,6 V (¡especialmente hoy día!), por lo que no es suficiente tensión para la EP 900.
Por este motivo hemos montado el adaptador con un conversor de tensión por condensador conmutado, muy flexible, que proporciona una tensión de salida regulada de + 5 V a partir de una tensión de entrada comprendida entre los valores de ¡2,7 y 5,5 V!. Sí, es correcto: el conversor trabaja exactamente igual tanto con una tensión de entrada más baja que la tensión de salida, como con una tensión de entrada más elevada que dicha tensión de salida, con una eficiencia que ronda el 90%. Hay que felicitar a los ingenieros de la casa Burr Borwn (la compañía que fue absorbida por Texas Instruments, lo cual explica por qué se tienen que obtener las hojas de características y especificaciones de la página web de TI). Sin embargo, la corriente que entrega este circuito está limitada a 30 mA.
El único inconveniente, desde el punto de vista de los aficionados, es el tamaño del circuito integrado del regulador (que sólo está disponible en su versión SM, es decir, Montaje Superficial), que hace que sea algo difícil de soldar. Afortunadamente, dicho componente tan sólo tiene seis terminales. Por lo tanto, es ahora o nunca el momento de probar nuestra destreza con un componente SM. Para ello, colocaremos el circuito integrado IC2 de forma precisa sobre sus puntos de soldadura. Aplicaremos un poco de estaño en uno de los terminales "+". Una vez que esta soldadura ha sido realizada correctamente y el componente no se ha movido, soldaremos la patilla diametralmente opuesta. Si todo está correcto y no hay ninguna patilla fuera de su punto de soldadura, pasaremos a soldar el resto de terminales. Si hemos creado un puente de soldadura entre dos terminales, podemos retirar el estaño sobrante con una trencilla de desoldar.

CONSTRUCCIÓN

Como se muestra en la **Figura 7**, la placa diseñada para este proyecto es de doble cara y utiliza tan sólo unos pocos componentes SM, principalmente en torno a la PLD EP 900. Naturalmente, estos componentes tienen que soldarse en la cara de pistas de la placa. Así que, ¡manos a la obra! Para ser lo más prácticos posible, recomendamos comenzar con los componentes SM. Debemos tener cuidado con algunos de ellos, en particular con el condensador C1, que se esconde en el centro de la placa, justo entre los terminales del zócalo PLCC 44 (sobre el que la EP 900 tiene que montarse por la otra cara). Tendremos cuidado a la hora de soldar el regulador IC2, ya que sin él, nada puede funcionar. Dicho componente está rodeado por condensadores que son mayores que el mismo. También tendremos cuidado en identificar correctamente los valores de los componentes SM (las resistencias tienen a menudo la información de su valor codificado: 103 significa 10 KΩ, 1203 significa 120 KΩ,…). Las cosas son aún más difíciles con los condensadores, los cuales a menudo no vienen identificados o no son identificables. Una vez que los componentes SM han sido montados, podemos montar la tira de resistencias, el resto de componentes convencionales, el conmutador SW, los conectores tipo "pinheader" K2 (MSP 40) a K5 (Altera), el zócalo PLCC 44 y, por último, el conec-

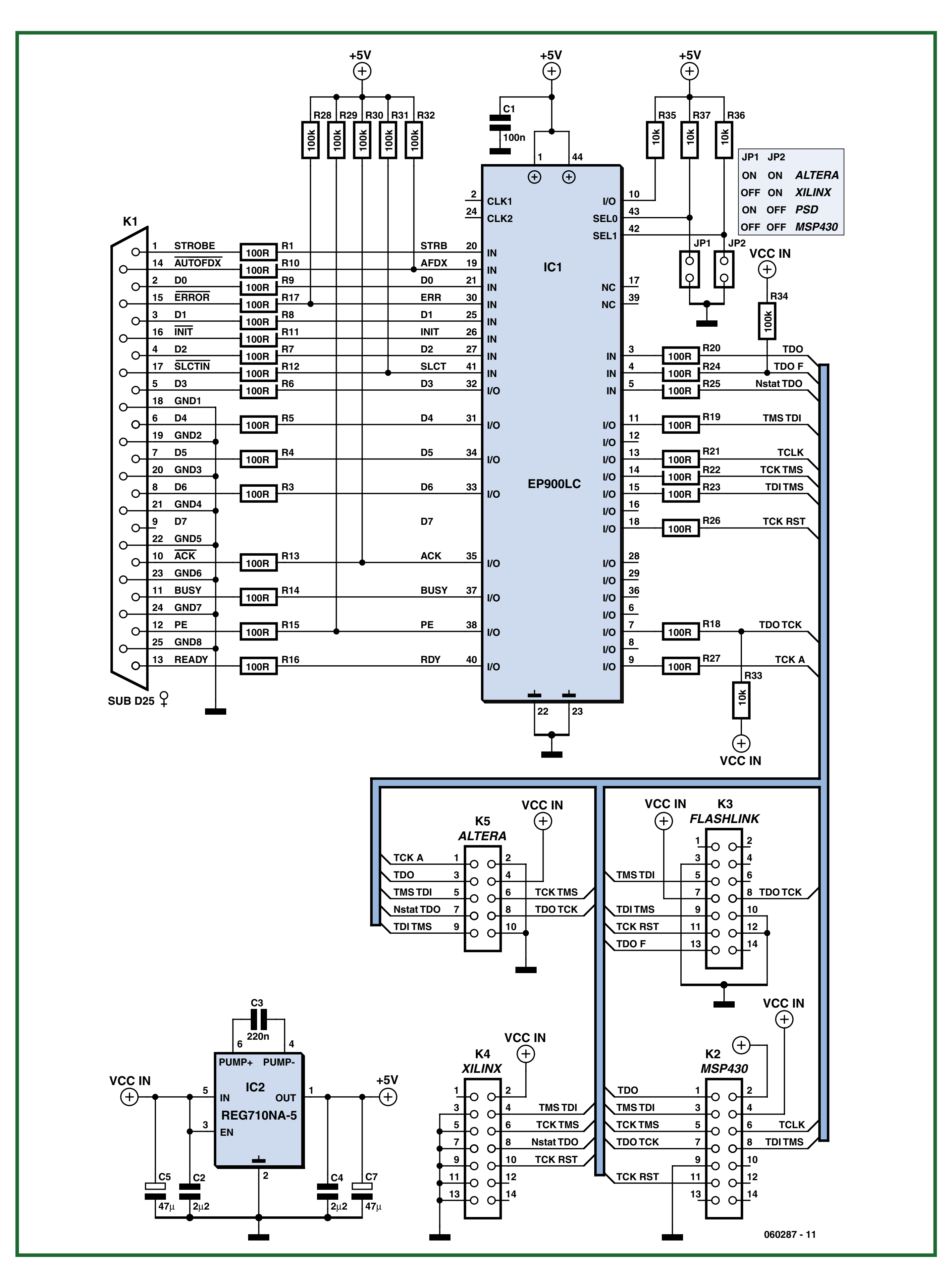


Figura 5. La EP 900 ocupa una buena cantidad de espacio en el centro del circuito para el programador JTAG universal. Este componente está disponible ya programado, de forma gratuita, cuando pedimos una placa de circuito impreso con código de pedido 060287-1.

tor Sub-D de 25 terminales K1. Debemos asegurarnos que hemos elegido la versión "macho" del conector de impresora (LPT), ya que la versión hembra no nos permitirá realizar una buena conexión. Una pequeña nota sobre el selector dual SW: no siempre es fácil conseguir un conmutador DIP dual, por lo que hemos dejado espacio para uno de cuatro posiciones, pero necesitamos cortar los terminales que nos sobran antes de montarlo sobre la placa de circuito impreso (PCI).
Si estamos construyendo nuestra propia placa, también es posible fabricar dicha placa en la forma de una sola cara, ya que la segunda cara de la placa de doble cara sólo se utiliza, de hecho, para evitar la necesidad de tener que colocar enlaces con hilos que requeriría la versión de placa de una sola cara. El montaje de los componentes es el mismo, pero en este caso es preferible, por razones prácticas, comenzar a soldar los enlaces con hilos, utilizando hilo de cobre estañado.
Deberemos poner cuidado en evitar los cortocircuitos con los enlaces con hilos colocados entre los conectores "FLASHLINK" y "XLINX", los cuales están relativamente cerca el uno del otro.
Todo lo que queda es conectar la PLD EP 900 sobre su zócalo. Verificaremos la calidad de nuestro montaje una última vez (soldaduras, valores de componentes…, afortunadamente sólo hay un valor para las resistencias convencionales), ya que no hay ninguna otra forma de verificar el funcionamiento correcto de este circuito excepto haciéndolo funcionar en modo real.
Nota sobre la PLD EP 900 (con código de pedido 060287-41: este componente está disponible ya programado, libre de gastos (aparte de los gastos habituales de manipulación y envío) en la tienda de Elektor. Si hacen el pedido de la placa de circuito impreso con la referencia # **060287-1**, el circuito integrado ya programado será suministrado automáticamente con ella.

ALTERA
D0 TRI TCK
D1 TRI TMS
D2 TRI PIN8
D3 TRI PIN6
D6 TRI TDI
AFDX
D4 WIRE ACK
TDO WIRE BUSY
PIN7 WIRE READY
WIRE ERROR

XILINX
D0 TRI TDI
D1 TRI TCK
D2 TRI TMS
D3 NOT
D6 WIRE BUSY
WIRE PE
VCC WIRE ERROR
WIRE ACK
D4 TDO AND2 READY

FLASHLINK
D0 TRI TCK
D1 TRI TMS
D2 TRI TDI
D3 NOT
D5 NOT RSTN
D6 WIRE READY
TDO NOT PE
WIRE ACK
WIRE BUSY
WIRE ERROR

MSP430
D0 TRI TDI
D1 TRI TMS
D2 TRI TCK
TDO TRI PE
SLCT NOT
INIT WIRE TEST
STRB TRI RST
AFDX NOT
WIRE ACK
WIRE BUSY
WIRE READY
WIRE ERROR

060287 - 16

Figura 6. ¡Nada como esto para ilustrar la flexibilidad de una PLD como la EP 900! Un único dispositivo puede realizar plenamente varias funcionales lógicas complejas.

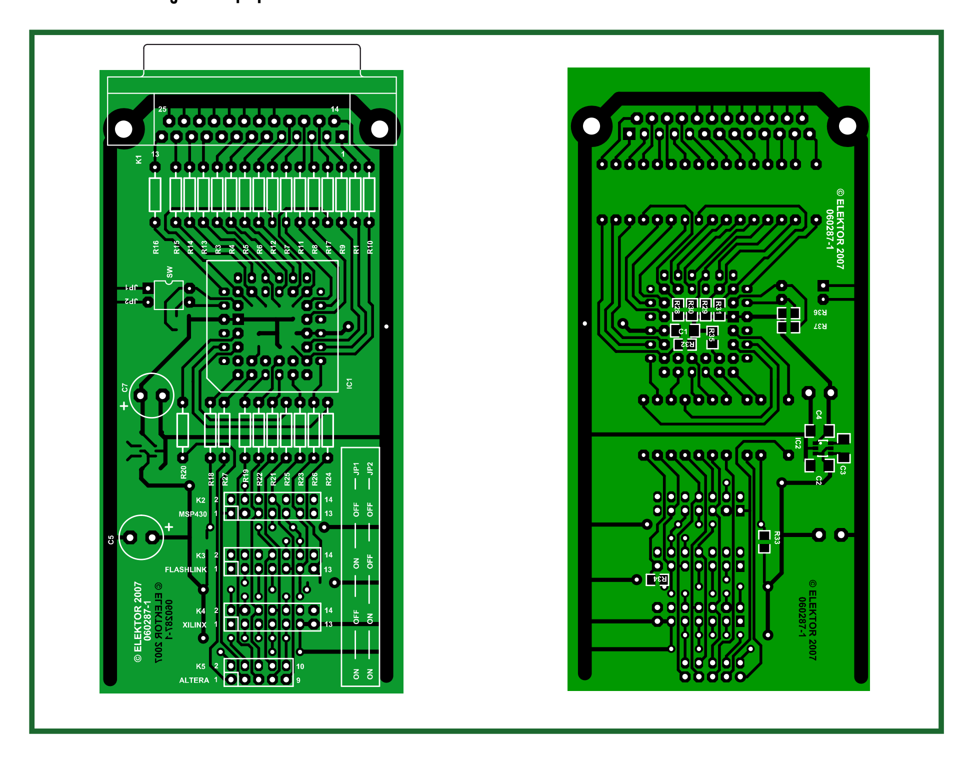


Figura 7. Planos de distribución de componentes para la placa diseñada para este proyecto. Los pasos para la creación de la placa de circuito impreso están disponibles para su descarga gratuita.

CONEXIONES DE DESTINO

¡Atención: tan sólo podemos utilizar un conector cada vez! En la mayoría de los casos un cable plano de 10 hilos o de 14, con un conector "crimpado" en cada extremo (tendremos cuidado en la orientación correcta de dichos conectores) es lo que necesitamos para establecer el enlace eléctrico entre el dispositivo destino y el adaptador (ver los esquemas eléctricos de los circuitos entre las Figuras 1 a 4 y la asignación de terminales en los correspondientes conectores).
Si tenemos un acceso directo a la parte trasera de nuestro ordenador, el adaptador puede insertarse directamente sobre el puerto LPT del mismo, sin necesidad de utilizar ningún cable adaptador que realice la interconexión entre el ordenador y el adaptador JTAG.

Adaptadores USB

Los ordenadores portátiles y de sobremesa actuales ya no disponen de puertos paralelos (LPT) (¡una decisión lamentable, especialmente para este proyecto!). Para dar solución a este problema, podemos intentar localizar un adaptador USB/LPT, pero debemos asegurarnos y verificar su compatibilidad con nuestro programador JTAG, ya que muchos de estos adaptadores que existen en el mercado tan sólo aceptan impresoras. No vamos a entrar en los detalles de los procedimientos de programación para todos los posibles dispositivos de destino, por lo que nos limitaremos a realizar sólo un ejemplo, el…

Cartucho ECG de la GameBoyi

El cartucho utiliza un conector SMD con una separación entre terminales de 1,25 mm (K3). Para fabricar el cable, recomendamos seguir los siguientes procedimientos:

1. Presionamos en la pieza de cable plano de 14 hilos sobre un conector DIL14 hembra.
2. Utilizar el conector Molex y los hilos ya preparados en la lista de componentes (ver número de Elektor de Diciembre de 2006) para realizar el conector adecuado de seis terminales para K3.
3. Soldar los cuatro hilos: TCK, TDI, TDO y TMS y los dos hilos de la fuente de alimentación a ambos conectores.
4. Verificaremos las conexiones con un medidor de continuidad y, a continuación, aislaremos la soldadura con un trozo de material termo-retráctil.

¡Ya lo hemos conseguido! Todo está listo para programar la PSD 813 en el cartucho ECG de la GameBoy.

Un último detalle: el adaptador es compatible con Byteblaster II (de la casa Altera), pero no trabaja correctamente con la primera versión de dicho controlador (Byteblaster sin más, sin los "II"). Este viejo controlador era utilizado por el programa MaxPlus II y ha sido sustituido por el programa Quartus hace dos o tres años.

(060287-I)

Lista de materiales

Resistencias
R1,R3-R27 = 100Ω
R28-R32,R34 = 100kΩ (SMD)
R33,R35,R26,R37 = 10kΩ (SMD)
(R2 no montada)

Condensadores
C1 = 100nF (SMD 1206)
C2,C4 = 2µF2 (SMD 1206)
C3 = 220nF (SMD 1206)
C5,C7 = 47µF condensador electrolítico de 10V radial
(C6 no montado)

Semiconductores
IC1 = EP900LC (programado, con código de pedido 060287-41)*
IC2 = REG710-NA5

Varios
K1 (K_LPT) conector Sub-D de 25 terminales macho, con terminales en ángulo recto para montaje en placa de circuito impreso (PCI)
K2 (FLASHLINK), K3 (MSP430), K4 (XILINX) conector tipo "pinheader" de 14 terminales en dos filas
K5 (ALTERA) conector tipo "pinheader" de 10 terminales en dos filas
J1,J2 (SW) conmutador DIP de 2 filas
Zócalo PLCC-44
Programa del proyecto en fichero # 060287-11.zip, descarga gratuita de la página web de Elektor
PCI con código de pedido 060287-1
* PLD ya programada suministrada cuando se pide la PCI # 060287-1 en la Tienda de Elektor

Opciones
Componentes para la conexión del cable a K3 sobre la GBECG:
- Zócalo IDC de 14 terminales (2x7) de presión
- Zócalo Molex, de 6 terminales, separación entre terminales de 1,25 mm (RS Components # 279-9178)
- 6 hilos con contactos crimpados para conector Molex (RS Components # 279-9544)

Bibliografía y Enlaces en Internet

[1] https://www.altera.com/support/software/download/sof-download_center.html

[2] http://www.xilinx.com/ise/logic_design_prod/webpack.htm

[3] http://focus.ti.com/docs/toolsw/folders/print/iar-kickstart.html

[4] http://mcu.st.com/mcu/modules.php?name=Content&pa=showpage&pid=57

Hojas de características con REF710-5 en:

http://focus.ti.com/lit/ds/symlink/reg710-5.pdf

Información suplementaria: fichero # 060287-11.zip, descarga gratuita desde: www.elektorelectronics.co.uk

Fichero fuente "AHDL" para el EP 900

Contrariamente a las primeras impresiones, un fichero AHDL puede informarnos sobre una gran cantidad de cosas. Mirando este fichero un poco más detenidamente es fácil diferenciar las distintas opciones (->).

```
subdesign prog_jtag_univers
(
TDO,Nstat_TDO,TDO_F : input;
STRB,AFDX,INIT,SLCT : input;
D[6..0] : input;
SEL[1..0] : input; -- 0->ALTERA,1->XILINX,
-- 2->FLASHLINK,3->MSP430
ACK,BUSY,READY,ERROR: output;
TCK_A,TMS_TDI,TCK_TMS,TDO_TCK,TDI_TMS,TCK_RST,PE : bidir;
)
variable
TCK_A,TMS_TDI,TCK_TMS,TDO_TCK,TDI_TMS,TCK_RST,PE : tri;
begin
TCK_A.in=D0; TCK_A.oe=AFDX;
case SEL[] is
when 0 -- ALTERA
=> TMS_TDI.in=D1 ; TMS_TDI.oe=AFDX;
TCK_TMS.in=D3 ; TCK_TMS.oe=AFDX;
TDO_TCK.in=D2 ; TDO_TCK.oe=AFDX;
TDI_TMS.in=D6 ; TDI_TMS.oe=AFDX;
TCK_RST.in=GND; TCK_RST.oe=GND;
ACK =D4;
BUSY =TDO;
PE.in=GND; PE.oe=GND;
READY=Nstat_TDO;
ERROR=GND;
when 1 -- XILINX
=> TMS_TDI.in=D2 ; TMS_TDI.oe=!D3;
TCK_TMS.in=D1 ; TCK_TMS.oe=!D3;
TDO_TCK.in=GND; TDO_TCK.oe=GND;
TDI_TMS.in=GND; TDI_TMS.oe=GND;
TCK_RST.in=D0 ; TCK_RST.oe=!D3;
ACK =GND;
BUSY =D6;
PE.in=D6; PE.oe=VCC;
READY=Nstat_TDO & D4;
ERROR=VCC;
when 2 -- FLASHLINK
=> TMS_TDI.in=D2 ; TMS_TDI.oe=!D3;
TCK_TMS.in=GND; TCK_TMS.oe=GND;
TDO_TCK.in=!D5; TDO_TCK.oe=VCC;
TDI_TMS.in=D1 ; TDI_TMS.oe=!D3;
TCK_RST.in=D0 ; TCK_RST.oe=!D3;
ACK =GND;
BUSY =GND;
PE.in=!TDO_F; PE.oe=VCC;
READY=D6;
ERROR=GND;
when 3 -- MSP430
=> TMS_TDI.in=D0 ; TMS_TDI.oe=!SLCT;
TCK_TMS.in=D1 ; TCK_TMS.oe=!SLCT;
TDO_TCK.in=D2 ; TDO_TCK.oe=!SLCT;
TDI_TMS.in=INIT; TDI_TMS.oe=VCC;
TCK_RST.in=STRB; TCK_RST.oe=!AFDX;
ACK =GND;
BUSY =GND;
PE.in=TDO; PE.oe=!SLCT;
READY=GND;
ERROR=GND;
end case;
end;
```

Para más información, el fichero de programación 'Jedec' (prog_jtag_univers.jed) está disponible en la página web de Elektor (www.elektor-electronics.co.uk).

Reflow Control

Soldadura de SMD con un horno eléctrico corriente

Paul Goossens

El laboratorio de Elektor, como muchos lectores, cada vez hace más soldaduras de componentes SMD. En la edición de enero de 2006 ya describimos en detalle como se puede construir un horno de reflujo con ayuda de un pequeño horno eléctrico económico. Este artículo tuvo mucho éxito entre los lectores, y nos dimos cuenta de que este tipo de proyectos genera un gran interés. En esta edición presentamos una versión totalmente renovada de la electrónica de control para hacerse un horno SMD. Además, se puede adquirir como kit de montaje, ¡de modo que el montaje será pan comido!

Tras más de dos años, nuestro horno de reflujo todavía trabaja con cierta frecuencia en nuestro laboratorio. Con motivo de las múltiples respuestas de nuestros lectores, nos pareció una buena idea hacer una versión renovada de este proyecto, que además se puede obtener como kit de montaje. De este modo, la autoconstrucción de un horno de reflujo para SMD será posible para todos.

Para los que se perdieron la primera versión de este artículo, examinaremos primero el proceso de reflujo.

¿El fin del soldador?

La soldadura convencional se realiza con un soldador. Primero hay que calentar las piezas que queremos soldar con el soldador. En cuanto la temperatura es suficientemente elevada, se añade un poco de estaño para soldadura, que se fundirá con ambas piezas. El resultado es una buena soldadura (según se espera).

Este método es bastante ideal para piezas, porque las conexiones se calientan una por una y la temperatura del chip permanece bastante baja.
Pero este método tiene una limitación, y es que hay que poder llegar a las dos piezas que deben soldarse con la punta del soldador. En varios SMD esto es complicado, ¡y a veces incluso imposible!

Reflujo

Uno de los métodos para soldar este tipo de componentes es la "soldadura por reflujo". En vez de soldar piezas calentando un soldador, con este método se calienta toda la tarjeta de circuito impreso, incluidos todos los componentes.
Tampoco se usa el estaño de soldadura estándar, sino que usamos lo que se llama pasta de soldadura. Este material gris consta de partículas sumamente finas de estaño mezcladas con líquido (o fundente). Esta pasta se coloca en la placa de circuito impreso primero. Después, se colocan encima los componentes. Luego se coloca todo en un horno, donde se calienta hasta el punto en que el estaño se funde con las capas de metal del entorno.

Perfil de temperatura

Parece muy fácil, así que ¿por qué necesitamos el circuito de control "Reflow-Control"? ¡Un horno normal también podría realizar esta tarea!
El truco está en que el circuito impreso debe recorrer un trayecto de temperatura concreto para obtener un buen resultado final.
Empezamos el proceso con el período de "Pre-heat" (precalentamiento). Aquí, la temperatura del horno se eleva a unos 125 °C. A partir de esta temperatura el fundente se licua. Este líquido fluye hacia los laterales, pero las bolitas de estaño permanecen en su sitio original.
Luego, la temperatura se eleva lentamente hasta un valor de 175 °C. Esta



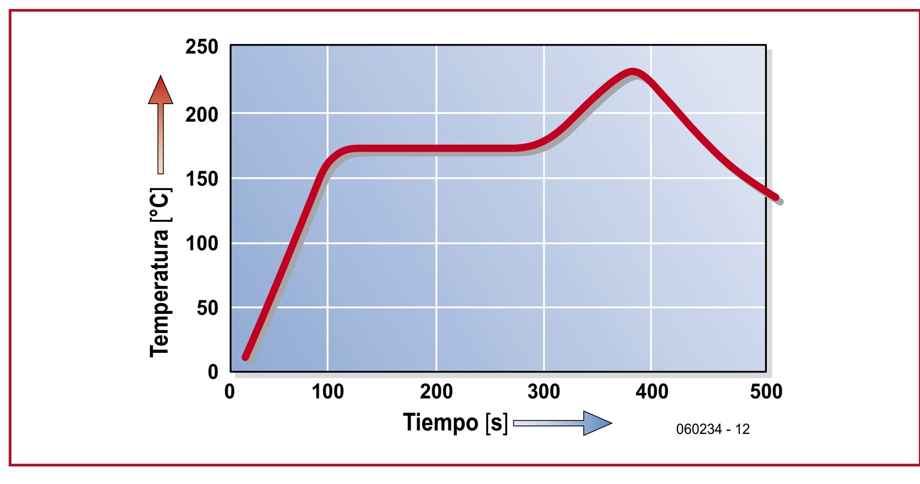


Figura 1. Perfil de la temperatura en el horno.

temperatura se mantiene en y cerca del punto de fusión de las bolitas de estaño. El motivo por el cual debe subir lentamente, es que la tarjeta de circuito impreso y los componentes deben tener tiempo para llegar todos aproximadamente a la misma temperatura. En términos de reflujo, denominamos a este paso "soak" (impregnación). Una vez alcanzada esta temperatura, el horno debe calentar la tarjeta y los componentes hasta una temperatura máxima (220...240 °C). Durante esta fase (que lleva el nombre de "reflujo")

Figura 2. Esquema del circuito de Reflow Control.

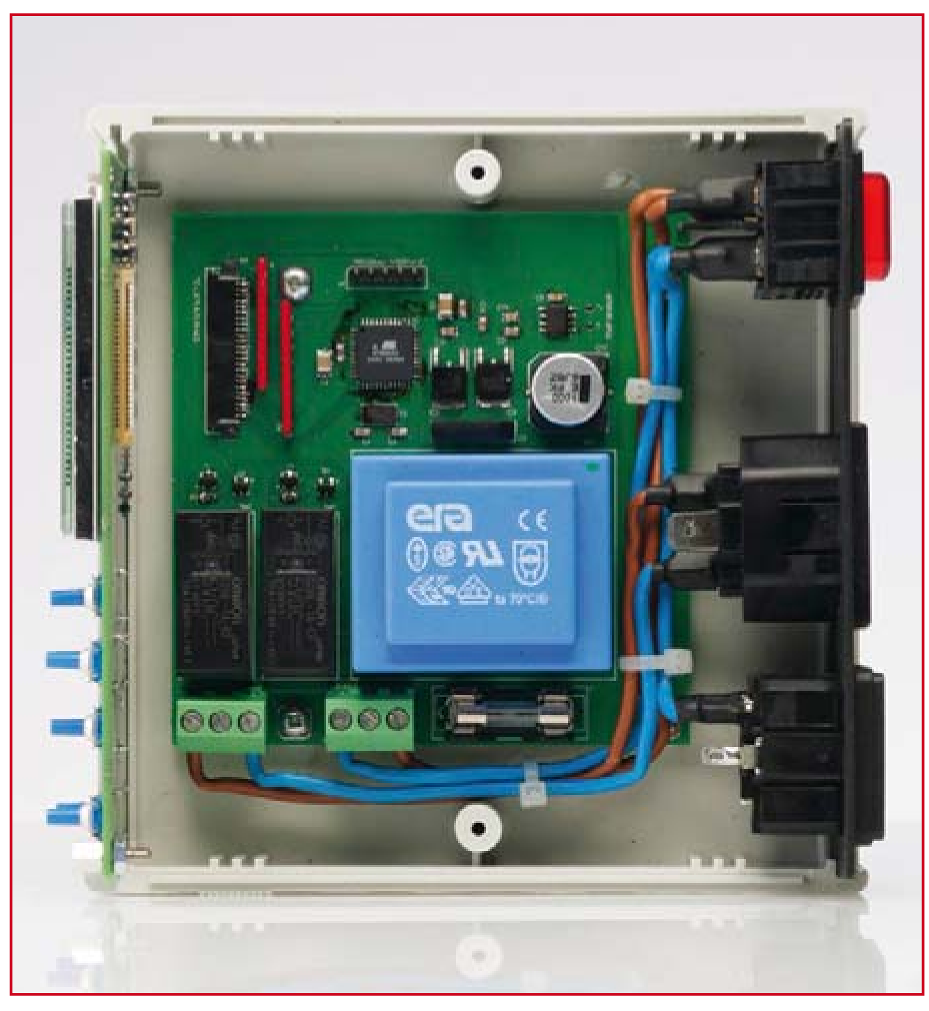


Figura 3. Vista del interior de la caja.

las bolas de estaño se derriten y se funden en las piezas de metal circundantes. La soldadura ya está hecha.
Después de alcanzar la temperatura máxima, debemos dejarlo enfriar todo. Esta fase, la denominamos muy originalmente “cooling” (enfriamiento). Este enfriamiento no debe producirse rápidamente, también para evitar grandes diferencias de temperatura entre los componentes y la tarjeta de circuito impreso. ¡En caso contrario, podrían deformarse o romperse!
Pero este enfriamiento, sobre todo al principio, tampoco debe ser muy lento. ¡Algunos componentes sólo pueden resistir un intervalo de tiempo máximo determinado por encima de una temperatura crítica!

Con o sin plomo

La pasta de soldadura obtenible se puede dividir más o menos en 2 grupos, según la composición de las bolas de estaño. Aunque hasta aquí hemos utilizado el término “estaño”, no es totalmente correcto. El metal de la pasta de soldadura es una aleación de dos o más metales.
En el caso de la pasta con plomo se trata en general de una aleación de estaño y plomo (SnPb). Esta aleación tiene el punto de fusión a 183ºC.
El segundo grupo, la pasta sin plomo, consta en general de una aleación de estaño, plata y cobre (SnAgCu). Por tanto no contiene plomo y el punto de fusión es más elevado. Hacia 217ºC esta aleación cambia a un estado más o menos fluido.
Al soldar con este último grupo, la temperatura máxima de la soldadura por reflujo debe alcanzar aproximadamente 240 ºC.
En la pasta con plomo bastará con 220 ºC, a veces incluso basta con una temperatura máxima de 200 ºC. Esto no sólo nos ahorra tiempo, sino que también permite que los componentes sufran menos durante el proceso de soldadura.
Conforme a las nuevas directivas RoHS las pastas con plomo ya no se usan para fabricar aparatos domésticos, con la única excepción de la automoción. Pero en el laboratorio se siguen usando estas sustancias con plomo para producir prototipos. Mientras sigan vendiéndose, haremos como si tal cosa.

Nuevo proyecto

Tras la descripción del concepto “reflujo”, llega el momento de estudiar detenidamente el circuito controlador.
Como queríamos hacer también para este proyecto un kit de montaje, resultó una buena ocasión para mejorar el proyecto original. En sí, la versión anterior no estaba nada mal, pero hemos adaptado un poco la electrónica con motivo de nuestras propias experiencias con la versión anterior y naturalmente las observaciones y críticas de nuestros lectores.
Tras una sesión de ruegos y preguntas entre los usuarios, resultó que nadie usaba el puerto de serie. Este puerto permite ver el perfil de temperatura durante el proceso de reflujo en un pc. Así que en lugar de ello, sería más práctico poder ver esta curva en el visualizador de la electrónica de control.
En el nuevo proyecto hemos eliminado el puerto de serie y la EEPROM. Y lo hemos sustituido por un visualizador de 2x16 caracteres para tener así un bonito visualizador gráfico. En cuanto a costes, ha resultado ser un poco más económico y el producto final es un poco más bonito.
Otra adaptación es la sustitución de los dos relés de estado sólido por relés convencionales. Esto también se ha hecho para ahorrar costes.

Esquema

El esquema del controlador (**figura 1**) no ha cambiado mucho respecto al anterior. El eje del circuito es el controlador IC1, un AT89S8253. Aquí funciona a una frecuencia de 12 MHz.
K5 es un interfaz de programación, con lo cual es posible programar nuevo firmware en el controlador. Para ello basta un simple programador. El controlador del kit ya está programado, por tanto para la mayoría de usuarios esta conexión no será muy interesante.
La alimentación se hace a través de Tr1, D3, IC4 y IC5, más algunos componentes circundantes para la amortiguación. Respecto a la alimentación, debemos mencionar también que el kit de montaje está disponible en 2

versiones, una para 230 V y la otra para 115 V. La diferencia entre ambas versiones está en el transformador de red empleado.
La tensión de red entra a través de K9. Esta tensión se conecta a través del relé Re1 y Re2 que a su vez se controlan a través de los FET T1 y T2. El horno se conecta en el conector K8.
La medición de la temperatura se hace a través del termopar. Este se conecta a través de un cable y un jack telefónico en los puntos K1 y K2. IC2 se encarga de la medición real, este IC establece la tensión del termopar para poder leer una temperatura absoluta a través de P2.0, P2.1 y P2.2 del controlador.
La conexión de los pulsadores S1 a S6 es muy sencilla. Están directamente conectados con las entradas P1.2 a P1.7 del controlador. Hay que tener en cuenta que estas conexiones también deben utilizarse durante la programación. ¡Esto implica que durante la programación a través de K5 no se puede accionar ningún pulsador! La resistencia elevadora R2 se encarga de que la tensión en las entradas esté a 5 V en reposo.
El LCD gráfico tampoco tiene sorpresas. Además del interfaz LCD habitual, se necesitan varios condensadores. Todos juntos se encargan de la electrónica en el módulo LCD para las tensiones de ayuda requeridas.
Los FET T3...T5 atacan los LED para la iluminación de fondo. En nuestro caso hemos elegido una iluminación RGB. Al atacar los 3 FET con una señal PWM, podemos darle el color que queramos a la iluminación de fondo.

¿Qué horno?

Además del circuito de “Reflow-Control” naturalmente necesitamos un horno para construir nuestro horno de reflujo. Este horno debe ser necesariamente de tipo analógico, que esté provisto de un termostato mecánico y un reloj mecánico. Además, la temperatura debe poder ajustarse como mínimo hasta 225 °C, mejor un poco más.
También es práctico que el horno sea lo más pequeño posible. En este caso la temperatura del horno podrá subir más rápidamente, y será suficiente con una potencia de unos 1,5 kW.
Por regla general, cualquier horno para pizza o minihorno analógico de 1,5 kW servirá.
En esta versión del “Reflow-Control” ya no hace falta modificar radicalmente el horno. Tan sólo habrá que colocar un termopar dentro del horno. Esto se puede hacer perforando el lateral, o con una abrazadera en el interior. ¡Hay que vigilar que el termopar esté derivado separadamente del horno!
La 2ª intervención es la adaptación del enchufe. Debemos cambiar el cable de red eléctrica estándar por el cable de red eléctrica que suministramos. Éste lleva un conector hembra que se adapta a nuestro “Reflow-Control”.

Montaje y uso

El montaje del Reflow-Control es muy sencillo. El montaje se limita a atornillar los diversos componentes y a conectar los conectores. El manual correspondiente se suministra con el kit de montaje. También se puede descargar gratuitamente del sitio web de Elektor. Asimismo, describe la puesta en marcha del horno de reflujo.

(060234)

Enlaces

[1] : http://www.elektor.com

[2] : http://www.8052.com/visisp52/

Cocción

Los componentes pueden coger humedad del aire. Durante el funcionamiento normal esto no es grave. Pero en el proceso de soldadura de reflujo puede dar problemas.

La temperatura del chip supera el punto de cocción del agua. Entonces, la humedad del chip formará vapor, con lo cual la presión del chip subirá considerablemente. La consecuencia de ello puede ser que el chip explote como consecuencia de este vapor.

Si los componentes han adquirido humedad, es posible sacarla de los chips calentándonos por unas horas hasta una temperatura de unos 80 °C.

Esto es muy importante si se trabaja con pasta sin plomo. ¡La presión del vapor aumenta de forma cuadrática con la temperatura!

El Reflow-Control está equipado con una función especial con la que se puede llevar a cabo el proceso de cocción.

Pasta de soldadura

Además de dividir la pasta de soldadura entre con plomo y sin plomo, en el mundo de las pastas hay otra división posible, basada en el fundente aplicado. El fundente puede ser químico o bastante reactivo, con lo cual será necesario limpiar la tarjeta de circuito impreso tras la soldadura para eliminar el fundente que haya quedado en la tarjeta. También hay tipos de fundente que son más o menos conductores. ¡Así que atención!

Lo mejor es usar una pasta de soldadura en cuya etiqueta figure ‘No-Clean’. Esto no significa que el bote y/o el contenido no estén limpios, sino que el fundente aplicado puede dejarse sin problema en la tarjeta de circuito impreso. Esto no afectará a la tarjeta de circuito impreso y además el residuo no será conductor y no perturbará el funcionamiento del prototipo.

Otro punto de atención es el tamaño de las “bolas de estaño”. Cuanto más finas sean las bolas, mejor se podrán dosificar con una jeringa en la tarjeta de circuito impreso. Aquí rige la simple regla empírica: “¡Menos es más”!

Firmware

El firmware para este proyecto se puede obtener, como en la mayor parte de los proyectos de Elektor, gratis en el sitio web.

Como compilador se ha usado el C-compiler SDCC gratis.

Quien tenga deseos específicos, y quiere programarlos él mismo en el controlador, entonces además del compilador necesitará un programador.

Tenemos buenas experiencias con el software VisISP-52. El programador que tiene (se puede encontrar en el sitio web de visISP52 [2]) es fácil de montar en placa perforada.

Tilt-Gamepad

Actualización del mando de juegos con sensor de aceleración

Von Xin Wang y Marko Westphal

Los controles intuitivos son la nueva moda entre los jugadores desde que llegaron las Nintendo Wii y las Sony PS3. La interacción directa sobre el juego, posible gracias a los controladores de la Nintendo Wii o del mando de la nueva PlayStation 3, con su función de inclinación, entusiasma a millones de usuarios. Sólo para quienes juegan en el PC no hay hasta ahora nada equiparable. El mando de inclinación, que presentamos en este proyecto, permite también a los jugadores de PC disfrutar de una vivencia interactiva con control intuitivo. El gamepad de inclinación es especialmente apropiado para juegos en donde se conduce o vuela, y para los de aventuras.

El Tilt-Gamepad es un mando de juegos con sensores de inclinación. Ya no es necesario presionar los botones de dirección. Es suficiente inclinar el mando para dirigir los movimientos del objeto. Se emplea un sensor de aceleración para determinar el movimiento de inclinación. Para evaluar las señales de los sensores de inclinación se necesita un microcontrolador. La placa aquí utilizada emplea un sensor de Analog Devices y un ATmega8 como microcontrolador. Con esta placa se puede transformar un gamepad común en un tilt-gamepad. Las señales digitales de movimiento emitidas por las salidas del microcontrolador (arriba, abajo, izquierda, derecha) simplemente reemplazan a las señales del botón de dirección del gamepad.

El sensor

La pieza central del tilt-gamepad es un sensor de aceleración analógico que suministra una tensión de salida proporcional a la inclinación. Para el proyecto, se escogió el ADXL322 de Analog Devices. Este sensor de aceleración, de dos ejes, trabaja con una tensión de operación de 2,4 V a 6 V. Cuenta con dos salidas analógicas, cada una con una resolución de 420 mV/90 grados. El rango de medición está en ±2 g. Dado que el sensor viene con un encapsulado SMD CP-16, no se puede trabajar con un soldador convencional. Por esta razón, para este proyecto, la tarjeta se suministra con un sensor ya montado.

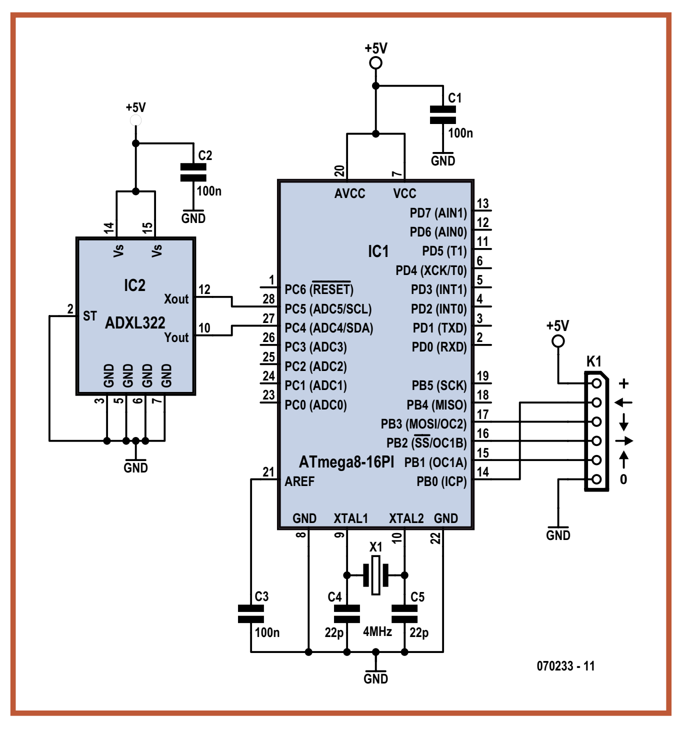


Figura 1. Aparte del sensor de aceleración y del microcontrolador, se necesitan pocos elementos.

Tratamiento de la señal

Ya que el sensor dispone sólo de salidas analógicas, es necesario un microcontrolador con puerto ADC para el procesamiento digital de la señal del sensor. Para satisfacer las exigencias, se ha seleccionado el microcontrolador de 8 bits ATmega8 de ATMEL. Cuenta con seis canales ADC de 10 bits y dos de 8 bits, al igual que con otros 13 pines de entradas y salidas.

Los niveles de tensión suministrados por el sensor de aceleración (valores de inclinación X-/Y-) pasan directamente al puerto ADC del microcontrolador y, a continuación, éste los digitaliza, procede a su filtrado y los prepara para la señal de control de salida. Las señales de salida emulan la función de los botones de dirección.

Los valores de X e Y se muestrean alternativamente, la ten-

sión de referencia de 2.56 V para el conversor A/D la produce el propio chip y se desacopla con el condensador C3 en el pin 21 (AREF). Los pines de E/S tienen capacidad para suministrar bastante corriente, lo que unido a la posibilidad de seleccionar resistencias de "pull-up", implica que no requiere amplificadores adicionales para las señales de salida.

Realización

Tal como lo indica el esquema de conexión de la Figura 1, el microcontrolador necesita solamente unos pocos componentes adicionales. Igualmente claro es el diagrama de la pequeña placa de dos caras de la figura 2.
El funcionamiento del software se ilustra en el diagrama de flujo de la Figura 3. El puerto ADC del microcontrolador se debe leer cada 10 milisegundos. A continuación, se convierte primeramente el valor de aceleración en inclinación y, después, éste se suaviza mediante un filtro. Dependiendo del sentido de la inclinación y de su ángulo, se suministran los respectivos valores a las salidas que van hacia el gamepad.
Las conexiones necesarias para la conexión al gamepad salen desde la regleta de conexión de seis polos K1. El suministro de energía se realiza directamente desde el interfaz USB (+5V y masa).

Transformación

Como objeto de transformación, se necesita un gamepad común. En principio, se pueden transformar todos los gamepads que tengan botones de dirección activos a nivel bajo. Los autores, por ejemplo, utilizaron el mando "Firestorm Digital 3" de Thrustmaster y, en el laboratorio de ELEKTOR, se utilizó el "MAXFIRE G-08X4" de Genius.
Para la transformación, aparte de la placa montada y el mando, se necesita un cable de señal de colores y seis hilos y una pequeña cubierta de plástico para la protección de la placa montada. No debemos insertar al principio el microcontrolador programado,. Se puede adquirir un microcontrolador ya programado o programar uno por cuenta propia. El código hex está disponible en la página web de ELEKTOR, como descarga gratuita. El código fuente no pudo ser proporcionado por los autores por los derechos relacionados con la licencia.
La transformación se inicia abriendo la parte trasera del gamepad, para lo cual se retiran todos los tornillos de la caja. A continuación, se debe averiguar dónde se encuentran las conexiones de +5 V y de masa. El método más sencillo es se-

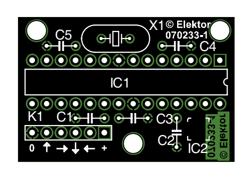


Figura 2. La placa de dos caras ya cuenta con un sensor montado. El sensor, tipo SMD, se encuentra en la parte inferior de la tarjeta.

Lista de componentes

Condensadores:
C1..C3 = 100 n
C4,C5 = 22 p

Semiconductores:
IC1 = Atmega8-16PI
(programado, EPS 070233-41*)
IC2 = ADXL322

Varios:
K1 = Tira de 6 pines.
X1 = cuarzo de 4 MHz
Placa con ADXL322 montado (EPS 070233-8191, ver ELEKTOR y www.elektor.es)
*El código hex está disponible como descarga gratuita en la web.

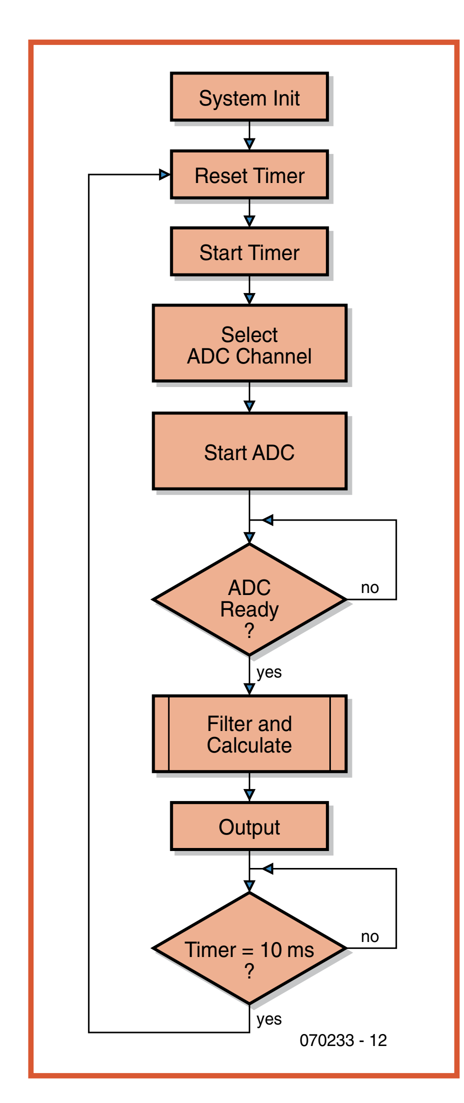


Figura 3. Diagrama de flujo del software evaluador. La exploración de los valores del sensor se realiza cada 10 ms.

guir el conductor USB. El Pin 1 es masa y, normalmente, es negro; el pin 4 es +5 V y, normalmente, es rojo. Procedemos de igual manera con los otros cuatro botones de dirección del gamepad. Bajo cada botón hay dos conexiones. Una de éstas casi siempre está conectada directamente a masa o a +5 V. En el Thrustmaster empleado como ejemplo, cada botón tiene respectivamente una conexión a masa. La otra conexión conduce directamente al microcontrolador. En esta conexión, posteriormente, se soldará un cable que estará conectado a las salidas del microcontrolador, en K1 (pines 2 a 5) de la placa. En el caso de que en el mismo mando no haya suficiente espacio para montar la tarjeta (como en el "Firestorm Digital 3"), se la alojará en una pequeña caja propia de plástico. Posteriormente, también en ésta, al igual que en la parte trasera del gamepad, se perfora un agujero de aprox. 5 mm de diámetro por el que se introducirá el cable.

Tabla. Cableado entre K1 de la placa del sensor y gamepad con cable de colores de seis hilos

K1	Tierra	Arriba	Izquierda	Abajo	Derecha	+5 V
Cable	Negro	Naranja	Amarillo	Verde	Azul	Rojo
Gamepad	Tierra	Arriba	Izquierda	Abajo	Derecha	+5 V

Ha llegado el momento de conectar los seis cables entre K1 de la tarjeta del sensor y el gamepad. Los cables se sueldan en +5 V, en masa y en los botones de dirección, en el lado del mando.

En el lado de la placa del sensor se indica la asignación de conexiones de K1 en el diagrama de conexiones (figura 1). Los pines 1 y 6 sirven para la alimentación de corriente. Los pines de 2 a 5 son salidas digitales para los botones de dirección (el nivel bajo es el activo, el estado normal es el nivel alto).

En la Tabla se ilustran claramente las conexiones que el cable de colores de 6 hilos establece entre K1 de la tarjeta y los puntos de conexión del mando. En la placa se pueden ver flechas junto a K1, que indican qué salida estará activa cuando uno mueva la placa en la dirección correspondiente.

Ahora se puede instalar el controlador programado en el zócalo de la placa. Después, se cierra cuidadosamente la caja de la placa, así como la del mando, y se adhiere a la parte inferior de la caja del gamepad con un pegamento termofusible.

¡Eso es todo! No cambia mucho la forma de jugar, pero, con la nueva función intuitiva de inclinación, se presenta una sensación complemente nueva, sin necesidad de controladores adicionales...

(070233e)

Enlace de Internet:
www.3dimotion.com (Website der Autoren)



USBprog

Herramienta universal de código abierto

Programa

Interfaz USB-R

Del licenciado en informática Benedikt Sauter, en colaboración con el doctor Thomas Scherer

¿Un nuevo microcontrolador y otra vez un programador nuevo? Hoy en día, los que se dedican a los controladores a menudo disponen de un arsenal de placas y adaptadores para la programación de diferentes chips. ¡El USBprog acaba con eso! Y como "extra" se utiliza como interfaz USB-I/O y USB/RS232.

Características y funciones:

- Interfaz USB
- Software para Windows y Linux
- Programador y depurador para procesadores AVR
- Programador y depurador para procesadores ARM
- USB para convertidor RS 3 sin driver
- Interfaz JTAG
- Interfaz USB-I/O (5 líneas)

A Benedikt Sauter, el desarrollador de este proyecto, también le molestaba que casi cada microcontrolador requiriera un programa propio. Eso significa espacio innecesario y no es ni económico ni ecológico. Además, muchos programadores sencillos de internet y productos comerciales antiguos requieren interfaces que ya no tienen los PCs modernos ni los portátiles. También ocurre con los interfaces paralelos de impresora, así como los interfaces RS232 serie que se utilizaban antiguamente. Los periféricos tienen que ser obligatoriamente conectables con un USB.
Para lo periféricos en serie ayudarían convertidores serie/
USB, pero a menudo son bastante lentos cuando el software también gestiona los cables de control de la interfaz en serie. La transmisión de algunos KBits a la memoria de un microcontrolador puede convertirse en una prueba de paciencia de un minuto, algo que en el desarrollo de software y sobre todo en la depuración pone nervioso. Sin embargo, los buenos (es decir, "rápidos") convertidores en serie/USB cuestan un poco más que la herramienta multiusos de MacGyver que les presentamos a continuación [1] para el mundo de los microcontroladores.

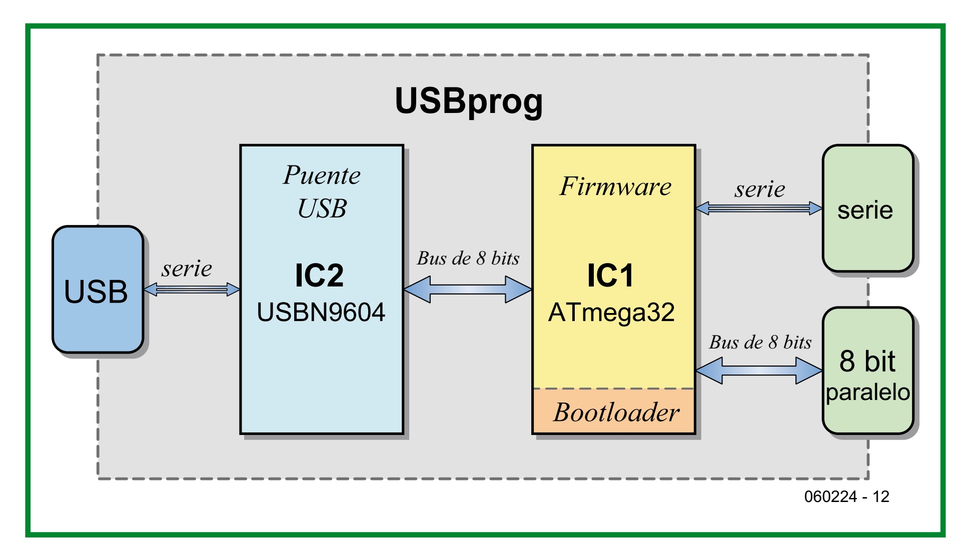


Figura 1. El diagrama de bloques del USBprog muestra la combinación de un puente USB con un microcontrolador.

¿Omnipotente?

Teniendo en cuenta esta situación de partida, tenemos al alcance de la mano la posibilidad de solucionar todos esos problemas de una vez por todas. Y es realmente posible:

Como para conseguir una transmisión de datos más o menos rápida entre el USB y el sistema de destino no tiene sentido colocar al lado de un chip de interfaz compatible con USB un microcontrolador independiente que asuma la comunicación rápida con el microcontrolador a programar, se puede recurrir a un dispositivo eficaz y utilizar un poco de inteligencia. Luego se pueden trasmitir los datos a la memoria del microcontrolador y éste se encarga luego del resto:

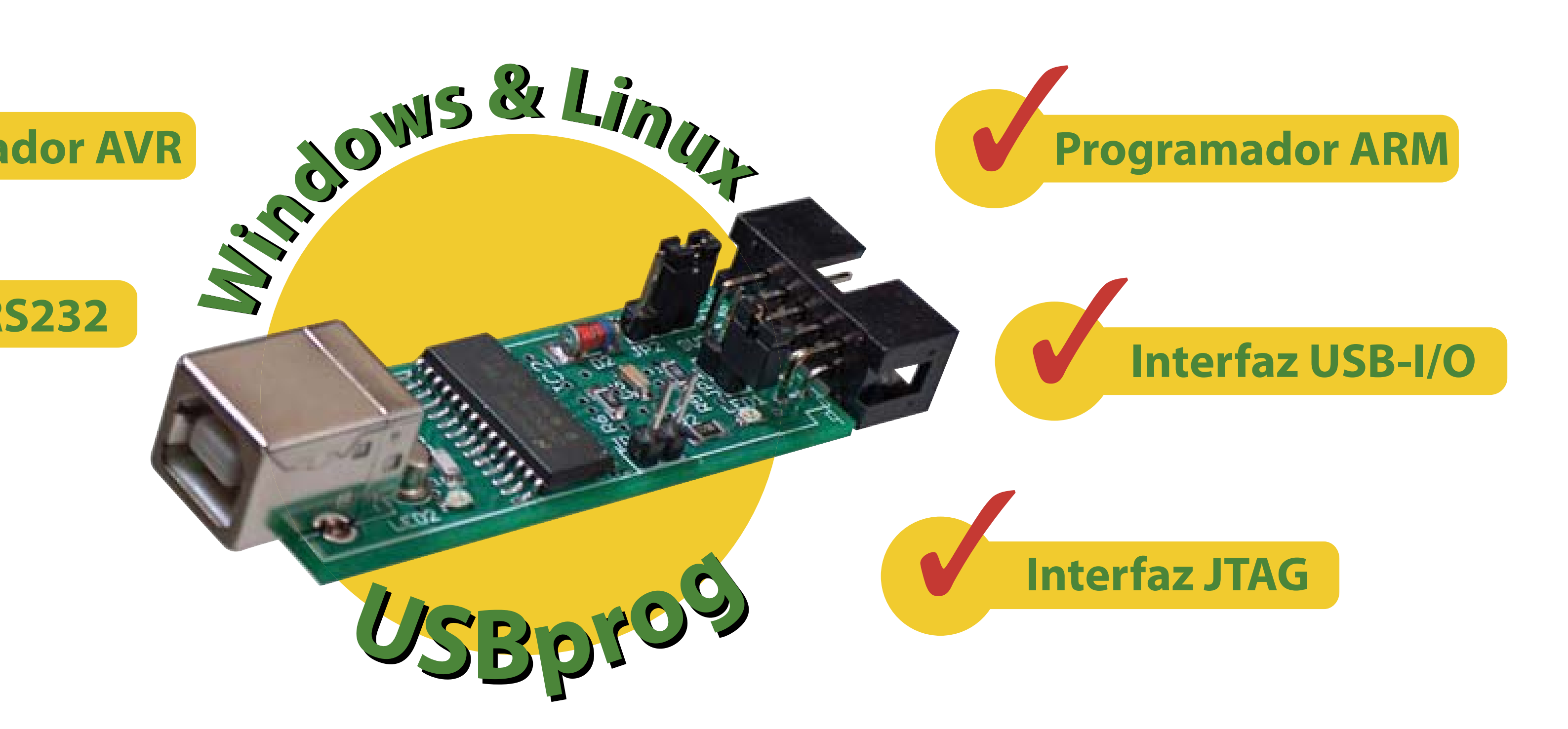


Tabla 1. Los firmware disponibles en la página del proyecto (08/2007, se ampliará continuamente)			
Nr.	**Firmware**	**Descripción**	**Estado**
1	Clon AVRISP-MK2	Copia del AVRISP-MK2 original	estable
2	Adaptador OpenOCD	Adaptador para ARM7 y ARM9 beta	beta
3	Programador AT89	Creado para el panel AT89S8252 de Elektor	beta
4	SimplePort	8 I/O navegables mediante C, Python y Java	estable
5	Adaptador JTAG	Adaptador universal USB-JTAG con biblioteca C	beta
6	Convertidor en serie/USB	Convertidor USB/RS232 (ifunciona sin drivers!) adicionales!)	pre-alpha
7	Clon JTAGICE-MK2	Copia del original JTAGICE mk2	Desarrollo (primeras funciones)
8	XSVF-USB-Player	XSVF Player universal	Concepto y texto fuente básico
9	Blink-Demo	Blink-Demo sencilla (el LED brilla)	estable
10	Programador PIC	Basado en Odyssey para Linux y Windows	Concepto y texto fuente básico
11	BDM-USB Interface	Interfaz, depurador de 68 procesadores	Concepto y estructura básica
12	MSP430-JTAG	Interfaz para controlador de ahorro de energía de Texas Instruments	Concepto y estructura básica

La simulación de un interfaz de programación y de la transferencia de datos al sistema de destino (véase **figura 1**). Pero será todavía mejor:

Este microcontrolador dispone de un cargador de arranque como programa básico, así que con el software adecuado del PC se le pueden cargar en todo momento en la memoria el firmware propio del microcontrolador directamente con el USB y sin un programador. Precisamente esa posibilidad ahora permite no sólo actualizar el firmware con comodidad, sino que sirve de vehículo para la herramienta universal. Así se pueden guardar firmware con las diferentes funciones en el microcontrolador del USBprog, que de ser un programador AVR pasa a ser un programador para el controlador ARM7/9 o en una interfaz de distribución USB. Y como el controlador que se utiliza en USBprog también lleva incorporado un UART, ¡también se añade la funcionalidad de un convertidor serial/USB gratis!

Código abierto

El hecho de que el dispositivo más bonito sin software no es más que una pieza electrónica inútil es especialmente cierto para el concepto de USBprog. Sin un cargador de arranque adecuado y sin firmware apropiados para todas las funciones imaginables se confirma esta funesta teoría. El que te adviertan de la posibilidad de escribir el software uno mismo tampoco ayuda mucho. Por supuesto, el desarrollador también lo tenía claro, por eso ofrece para descargar más que sólo los drivers necesarios para el funcionamiento del USB de IC2 y el cargador de arranque para IC1 en su página web del proyecto [2] y en www.elektor.com.

La **tabla 1** muestra una lista de firmware adecuados, que se amplía continuamente. Incluye datos muy interesantes que convierten el USBprog en un clon del programador dentro del sistema MK2 de Atmel. Así, se puede utilizar sin problemas el entorno de desarrollo AVR-Studio que ofrece Atmel de forma gratuita y otro software que

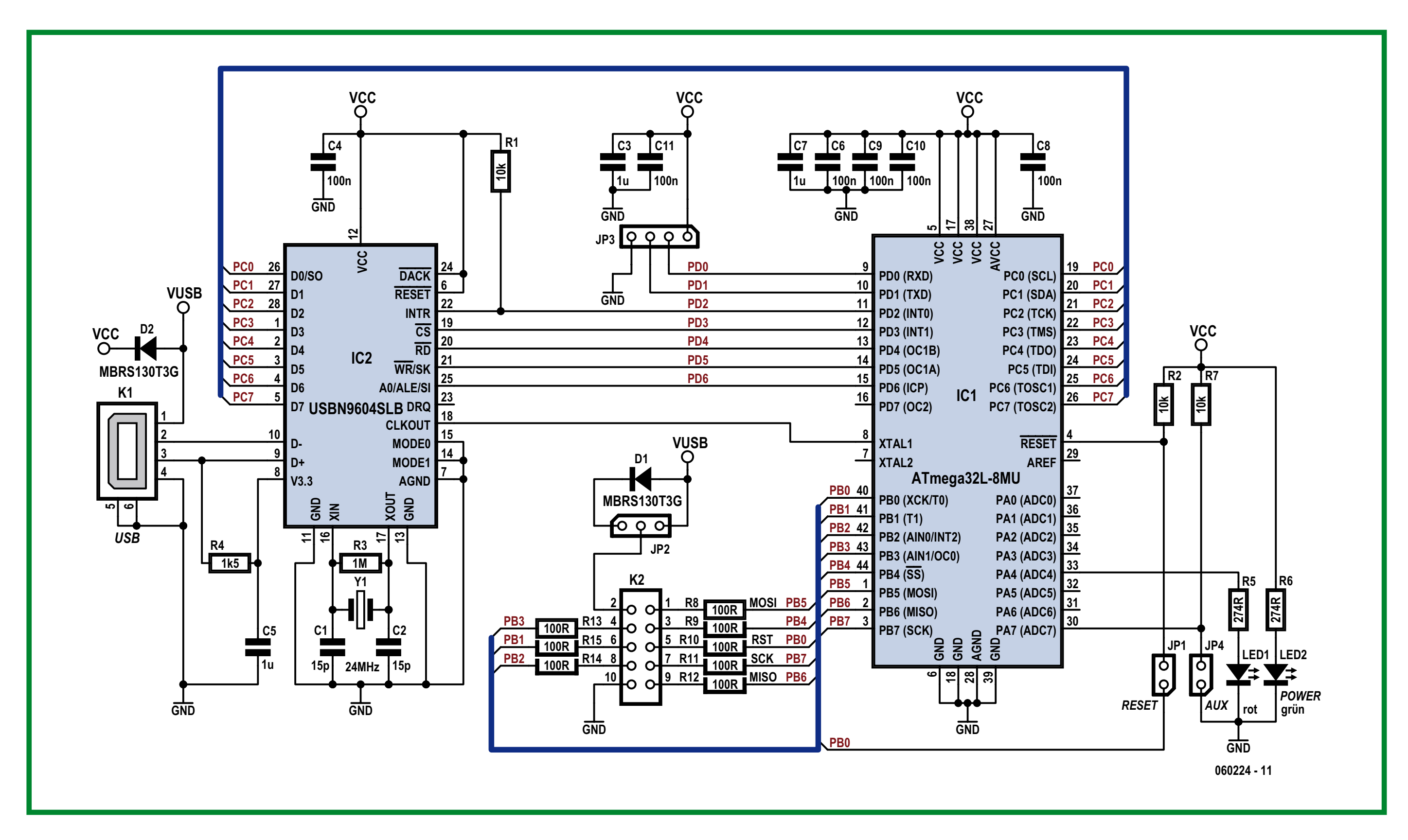


Figura 2. El diagrama de circuito del USBprog no es mucho más complejo que el diagrama de bloques.

soporta el AVRISP-MK2.

Otro firmware permite programar y depurar un controlador ARM7 o ARM9 con ayuda del entorno de software OpenOCD [4]. También está pensado para su uso como convertidor serie/USB o como un interfaz sencillo de USB con diez I/O digitales. Además de la función obligatoria de programador con interfaz JTAG, en estos momentos se está trabajando la compatibilidad con más familias de controladores.

Todo el software se presenta (en la línea de la mentalidad de código abierto) como texto fuente abierto. Todo el mundo está invitado a ampliar el fondo de posibles firmware por el bien de la comunidad. Ya existe un pequeño grupo de aficionados al USBprog. Sin embargo, para reducir el acceso que se produce en los proyectos de código abierto, a menudo demasiado alto, en la página web del proyecto hay disponible software que no sólo puede renovar el Atmega32, sino que apretando un botón en la página del proyecto se puede comprobar si han llegado nuevos firmware para él y descargarlos dado el caso.

Todo ello existe no sólo para Windows,

Tabla 2. Distribución de las clavijas SV según los firmware utilizados.

Pin Nº	Clon AVRISP-MK	Adaptador OpenOCD	Programador AT89	SimplePort	Adaptador JTAG	Convertidor serial USB	Clon JTAGICE-MK2
K2 #1	MOSI	TDO	MOSI	Port 1	TDO	–	TDI
K2 #2	VCC	VREF	VCC	VCC	VREF	VCC	VREF
K2 #3	–	SRST	–	Port 2	SRST	–	–
K2 #4	–	TRST	–	Port 3	TRST	–	–
K2 #5	Reset	TMS	Reset	Port 4	TMS	–	TCK
K2 #6	–	–	–	Port 5	–	–	–
K2 #7	SCK	TCK	SCK	Port 6	TCK	–	TMS
K2 #8	–	–	–	Port 7	–	–	–
K2 #9	MISO	TDI	MISO	Port 8	TDI	–	TDO
K2 #10	GND	GND	GND	GND	GND	GND	GND
JP3 RX	–	SRST	–	Port 9	–	RX	–
JP3 TX	–	TRST	–	Port 10	–	TX	–

también para Linux, y está pensada incluso una versión OS-X.

Dispositivo

La **figura 2** muestra el circuito del USBprog. No es mucho más complicado que el diagrama de bloques. El circuito completo se alimenta directamente desde el conector del USB del PC, algo que con una toma de corriente general de menos de 100 mA es posible sin problemas. D2 protege el PC que le da alimentación de posibles sobrecargas de tensión mediante el dispositivo cerrado de USBprog.

IC2 es un puente USB especializado que es lo bastante rápido para nuestro objetivo con un USB 1.1 “full speed“ compatible y su transmisión de datos basada en DMA. La frecuencia del cuarzo de 24 Mhz exterior se dobla en el interior y se puede poner en CLKOUT con una frecuencia programable, de manera que IC1 se vuelve cíclico, así que sobra un segundo cuarzo. La mayoría de firmware vuelven cíclico IC1 con su máxima frecuencia de 16 MHz. IC1 e IC2 se conectan entre sí mediante algunos cables de control y un bus ancho de 8 bits. A las clavijas de las salidas paralelas K2 y de JP3 se les puede agregar según las necesidades las funciones adecuadas (véase la **tabla 2**).

Si está introducido JP1, se puede controlar la transmisión de reinicio de IC1 desde fuera mediante el pin 5 de K2, imprescindible para la programación externa (cargador de arranque). Con JP2 se puede escoger si hay que alimentar un dispositivo cerrado K2 directamente de 5 V de la interfaz de USB o si el PC debe estar mejor protegido por D1. D1 y D2 son diodos Schottky por las mínimas caídas de tensión. JP4 está disponible para fines personales, se podría conectar, por ejemplo, un pulsador adicional.

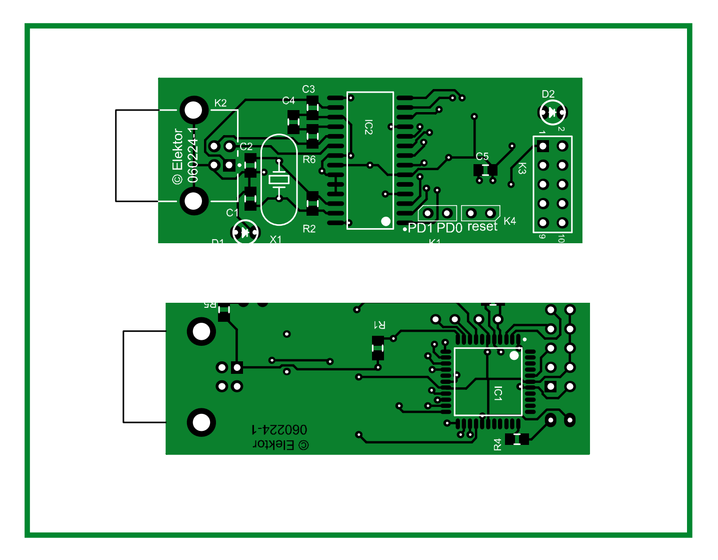


Figura 3. La placa de dos caras de tiene montaje SMD.

Estructura y cargador de arranque

Como el USBprog se hace con una placa de doble cara y algunos componentes SMD, seguro que muchos lectores recurrirán encantados en la tienda Elektor al kit disponible con la placa ya ensamblada. Sólo hay que soldar los conectores. Quien tenga ánimo para hacerlo, también puede fabricarse su propia placa (**figura 3**) con ayuda de los layout (descarga en www.elektor.com) o pedirla en www.thePCBShop.com y montarla completamente a mano.

Tras haberla montado con éxito (**figura 5**) se presenta un problema como el del huevo y la gallina: antes de poder escribir los firmware deseados en la memoria flash con comodidad con el USB, primero hay que guardar el cargador de arranque en IC1, que se puede hacer gracias al flash con USB. El puerto B de IC1 indica como segunda función las conexiones necesarias para la programación, y todas ellas se llevan a K2. Quien no pueda disponer de un programador AVR o tomarlo prestado de un amigo o conocido, tal vez tenga acceso a un PC con una interfaz paralela. Mediante esa interfaz de impresora se puede programar también IC1 con una conexión muy sencilla, rápida y unida [5].
El archivo “usbprog_base.hex“ se trasmite así con AVR-Studio, AVRDUDE u otro soft-

Lista de piezas

Resistencias:
R1, R2, R7 = 10 k, SMD 0603
R3 = 1 M, SMD 0603
R4 = 1k5, SMD 0603
R5, R6 = 274 Ω, SMD 0603
R8...R15 = 100 Ω, SMD 0603

Condensadores:
C1, C2 = 15 p, SMD 0603
C3, C5, C7 = 1 μ, SMD 0603
C4, C6, C8...C11 = 100 n, SMD 0603

Semiconductores:
IC1 = ATmega32L-AU, QFP 12x12 (Atmel)
IC2 = USBN9604SLB (National Semiconductor)
D1, D2 = MBRS130T3G, SMB DO214AA
D3 = LED rojo, SMD 0603
D4 = LED verde, SMD 0603

Además:
Q1 = Cuarzo, 24 MHz, HC49/4
X1 = Conector USB, tipo B para montaje en PCB
K2 = Conector DIL de 10 pines, para montaje en PCB
JP1, JP4 = Conector SIL de 2 pines
JP2 = tira de 3 pines
JP3 = tira de 4 pines
Placa 060224-1 (Layouts y pedidos en www.elektor.com)
Placa SMD montada con todos los componentes necesarios (060224-71, véase el escaparate de la tienda Elektor)

Figura 4. Una captura de pantalla de “usbprog-gui.exe” en Windows.

Sobre el autor

Benedikt Sauter es un apasionado desarrollador de software y hardware de código abierto, y en la página web del proyecto se ocupa de la ampliación y el fomento de las aplicaciones posibles con el USBprog.

Tabla 3. Fusibles aplicables al ATmega32

Nº	Fusible	Valor	Descripción
1	BODLEVEL	1	Sin Brown-Out-Detection
2	BODEN	0	Sin Brown-Out-Detection
3	SUT0	0	Startup-Time = 6 CLK + 64 ms
4	SUT1	1	Startup-Time = 6 CLK + 64 ms
5	CKSEL3..0	0000	Tiempo externo
6	CKOPT	1	Tiempo externo
7	OCDEN	1	On-Chip-Debug disable – importante por el puerto C
8	JTAGEN	1	JTAG fuera
9	SPIEN	0	serial program download enabled
10	EESAVE	1	Chip erase apaga también EEPROM
11	BOOTSZ1..0	00	Boot start address = $3800
12	BOOTRST	1	Sólo en "usbprog_base.hex" – si no BOOTRST = 0

ware AVR al ATmega32. Además los fusibles se deben colocar según la **tabla 3**.

Una vez esté el cargador de arranque y los fusibles colocados, el USBprog está disponible.

Software

Como ya hemos mencionado, hay una versión de línea de mando para Linux y Windows, así como un programa "correcto" para grabar en la memoria flash con comodidad de USBprog.

Para Windows hay en la página web del proyecto [2] un programa de instalación con la denominación "Installer.exe". Cuando se ejecute este archivo en Windows se instalará, además de la versión de línea de mando "usbprog.exe" y la versión con soporte gráfico "usbprog-gui.exe", el driver correspondiente. Se conecta el USBprog a un interfaz de USB, el asistente de drivers de Windows debería reconocer el USBprog y activar esos drivers automáticamente.

El software para Linux no se presenta como un programa ya compilado por la variedad de arquitecturas que existen. Como de costumbre, hay que descargar el archivo completo de software de [2] y proceder como le indicamos a continuación:

1. Desempaquetar con "tar xvzf usbprog.tar.gz".
2. Cambiar a la lista "usbprog".
3. Preparar la compilación con la orden "./configure"
4. Iniciar la compilación con la orden "make".
5. Instalar con la orden "make install" (¡ejecutar como "root"!)

De momento no existe ningún programa de flash USBprog para Mac, pero se puede utilizar un USBprog con flash como clon de AVRISP-MK2 con el programa gratuito "AVRDUDE" [6], un programa de línea de mando para microcontroladores de AVR.

En la página web del proyecto [2] siempre están actualizadas las novedades en USBprog.

(060244e)

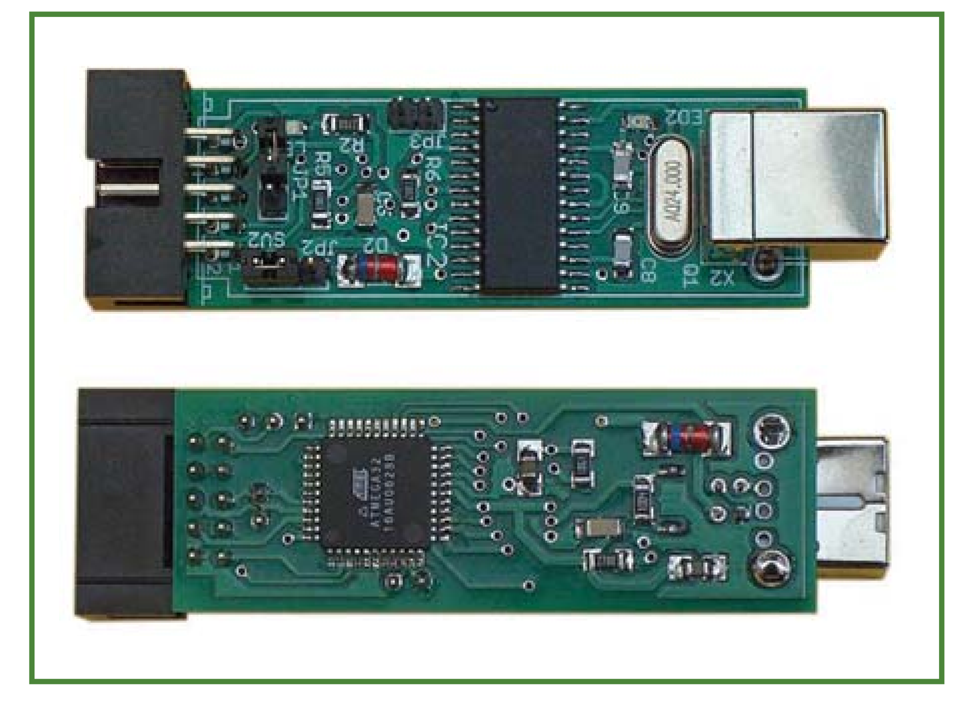


Figura 5. Este modelo construido en el laboratorio Elektor corresponde ya casi a la producción en serie…

Enlaces

[1] MacGyver:
www.hypediss.com/mac_gyver_multitool

[2] Página del proyecto:
www.embedded-projects.net/usbprog und bei www.elektor.de

[3] AVR-Studio:
www.atmel.com/avrstudio

[4] OpenOCD:
http://openocd.berlios.de/web

[5] Programador paralelo:
www.ixbat.de/index.php?page_id=188

[6] AVRDUDE:
www.nongnu.org/avrdude/

[7] WinAVR:
http://winavr.sourceforge.net/

[8] Yartago (entorno de desarrollo ARM):
www.yagarto.de/

[9] Ficha técnica ATmega3:
www.atmel.com/dyn/resources/prod_documents/doc2503.pdf

Analizador OBD-2



Interpreta "códigos de error" sin un ordenador

Folker Stange and Erwin Reuss

Este analizador portátil manual realiza el sencillo trabajo de "rebuscar" entre la información almacenada en la parte accesible al cliente del computador de nuestro coche. El dispositivo funciona con todos los protocolos OBD-2 actuales y puede leer y borrar códigos de error almacenados en el vehículo, al mismo tiempo que puede reiniciar la pantalla MIL. Todo ello sin la ayuda de un ordenador o de visitar el servicio técnico correspondiente.

Desde la llegada del nuevo milenio, cada vez más, los nuevos modelos de coche han sido equipados con la última versión de la interfaz de diagnostico de a bordo OBD-2. Con el incremento de la sofisticación en la gestión de los modernos motores, muchos nuevos propietarios han visto los beneficios de un analizador OBD, de manera que rápidamente se ha convertido en una parte esencial del kit de herramientas de su garaje, junto con la llave de bujía y la llave dinamométrica.
Se ha escrito que, algunas veces, cuando los propietarios instalan una nueva radio o sistema de navegación en su coche, el sistema de gestión del vehículo innecesariamente registra un fallo, de forma similar a algunos propietarios que modifican el motor para poder usar un combustible alternativo y han notado que la gestión del motor puede interpretar incorrectamente la nueva situación del motor y genera un error. En algunos casos la gestión del motor puede llegar a convertirse en una condición de emergencia. Cualquier que sea la causa el resultado es el mismo: un MIL (indicador luminoso de mal funcionamiento) montado en el salpicadero se enciende, se registra una condición de fallo y es necesario hacer una visita (cara) al taller más próximo para arreglar el "problema" y reiniciar el MIL. Con el analizador OBD descrito aquí en nuestra guantera, es un trabajo sencillo conectar el equipo al conector OBD de nuestro coche, encontrar de qué problema se trata, reiniciar el error y continuar viaje. Pensando en el coste básico, sólo el precio del analizador estará más que recuperado si evitamos una única e innecesaria visita al taller del distribuidor.
En las revistas de electrónica ya han sido descritos un cierto número de analizadores OBD (incluyendo Elecktor), donde se describe una interfaz entre el conector OBD del vehículo y un ordenador portátil. Sin embargo, la configuración que presentamos aquí es menos enrevesada. Esta unidad autónoma no requiere un ordenador portátil o una batería, reconoce todos los protocolos habituales OBD-2 o EOBD y es lo suficientemente pequeña como para guardarla en la guantera del coche. El manejo es bastante sencillo por medio de sólo dos botones, 580 de los códigos de los problemas más comunes se pueden reconocer y describir en su pantalla de texto.

2 autónomo

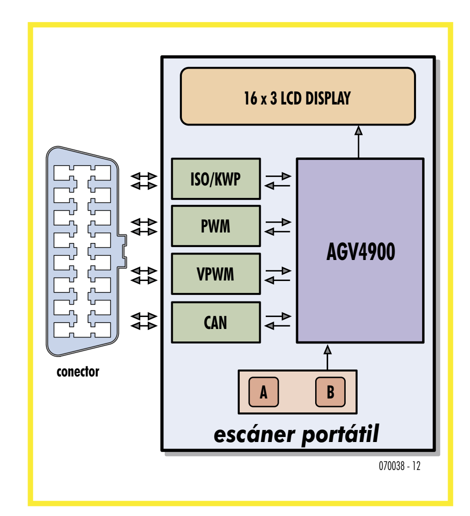


Figura 1. Diagrama de bloques del analizador OBD-2.

El circuito

El analizador OBD-2 emplea un microcontrolador AT-90CAN128 de la familia AT mega128 de la casa Atmel. Este particular modelo tiene equipado una interfaz para el bus CAN, tal y como se muestra en la **Figura 1**. El controlador se suministra pre-programado con el programa interno ("firmware") del AGV 4900, el cual maneja la interfaz de usuario, incluyendo botones, timbre, diodos LEDs y pantalla LCD.

La asignación de terminales en el conector OBD-2 se muestra en la **Figura 2**. Para poder soportar todos los protocolos OBD-2 actuales, el analizador necesita poder conectarse a varios interfaces bidireccionales:

- Interfaz K/L
- Interfaz PWM
- Interfaz VPWM
- Interfaz CAN

Los primeros tres de esta lista han sido implementados en el esquema eléctrico del circuito (**Figura3**) usando transistores y comparadores configurados para adaptarse a las interfaces estándar. Las resistencias especificadas de "pull up" para la señales K y L tienen una impedancia relativamente baja, por lo que los transistores MOSFETs se han usado aquí como controladores. El circuito integrado controlador del bus CAN es del tipo PCA

Especificaciones

- Selección automática o manual del protocolo OBD-2.
- Búsqueda automática muy rápida del protocolo (de 0,1 a 2,6 s por protocolo)
- Secuencia de arranque de programa muy rápida (listo para trabajar un segundo después del encendido).
- Lee y muestra información importante sobre el vehículo (dependiendo del coche).
- Lectura de sensores en tiempo real (seleccionable).
- Muestra el número de chasis del vehículo (si es soportado por el fabricante del vehículo).
- Lee y muestra el código de error almacenado en memoria.
- Lee y presenta en pantalla el dato de captura de trama.
- Borrado del código de error de la memoria.
- Selección de idioma (Inglés o Francés).
- 580 códigos de error con descripción de los mismos en texto deslizante.
- Están soportados todos los protocolos OBD existentes para vehículos privados:

ISO9141-2

ISO14230-4 (KWP2000)

J1850 PWM

J1850 VPWM

ISO15765-4 (CAN, 11/29 Bit, 250/500 kbits/s)

- La alimentación para el analizador se suministra del conector OBD-2 del vehículo (12 V).
- Pantalla LCD de tres líneas con iluminación de fondo y contraste ajustable.
- Señal acústica que proporciona unos tonos audibles cuando se reconoce un código de error.
- Diodos LED indicadores del estado de la conexión y del flujo del tráfico de datos.
- Funcionamiento sencillo utilizando tan sólo dos pulsadores.
- Conexión por medio de un cable OBD-2 estándar.
- Formato manejable y portátil: 80 x 135 x 30 mm (ancho x alto x fondo), peso 150 g (aprox.).
- Suministrado como KIT a través de Elektor SHOP.

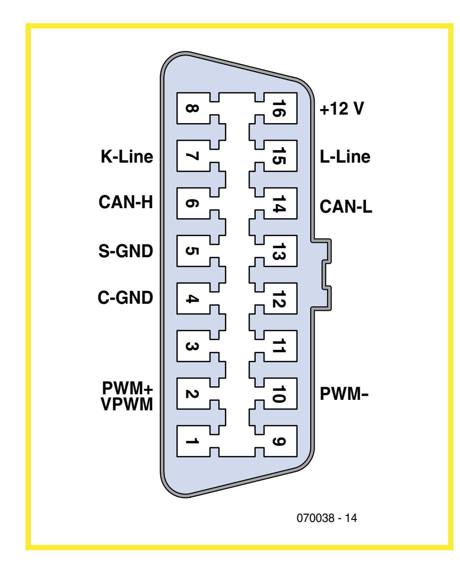


Figura 2. Identificación de terminales en el conector OBD-2 / EOBD.

Puntos a señalar

El analizador OBD sólo es adecuado para vehículos equipado con un conector OBD-2/EOBD.

El conector EOBD se monta sobre vehículos vendidos en la Unión Europea:

• Después del 01 / 01 / 2001, para vehículos con motor de gasolina.

• Después del 01 / 01 / 2004, para vehículos con motor de gasoil.

Antes de conectar el analizador a ningún vehículo fabricada antes de estas fechas, es importante verificar la compatibilidad de dicho vehículo con el estándar OBD-2. La página web de Florian Schäffer [3] contiene una base de datos de vehículos donde podemos comprobar si el nuestro es compatible con el estándar OBD-2.

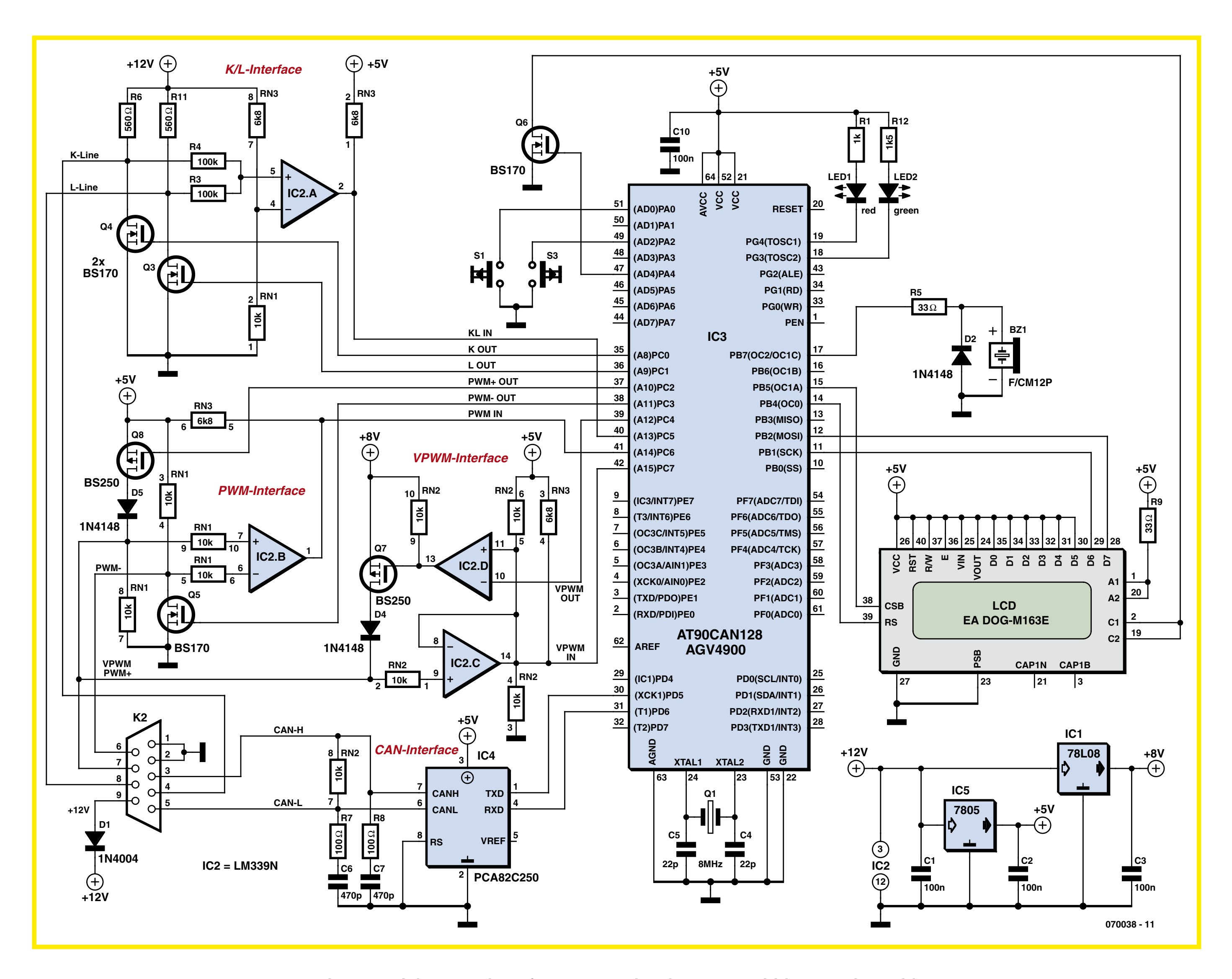


Figura 3. El microcontrolador AVR, con la interfaz CAN incorporada, es la parte principal del esquema eléctrico del circuito.

El programa interno

El corazón del analizador OBD-2 es el microcontrolador pre-programado con la designación AGV4900 [1] disponible solamente en la casa Stange Distribution. El programa ha sido desarrollado por el co-autor Erwin Reuss. Como otros proyectos similares OBD, el programa interno (“firmware”) para este analizador sólo está disponible pre-programado dentro del propio microcontrolador donde es copia protegida. El fichero fuente no está disponible para su descarga. Sin esta protección de copia del programa no sería posible ofrecer el analizador en forma de kit. No hay posibilidad para el montador casero de ensamblar una versión de bajo coste de este diseño a menos que, por supuesto, toda la programación necesaria sea escrita desde cero.

Una opción del menú cambia toda la información de la pantalla entre inglés o francés (por el interés de nuestros lectores canadienses). La casa Stange Distribution está especializada en el equipo OBD comentado y producen varios controladores OBD-2 para aplicaciones en el campo del desarrollo ODB.

Todas las funciones del analizador OBD pueden seleccionarse desde el menú usando tan sólo dos teclas. Una característica del programa es el rápido procedimiento de arranque, el cual asegura que el aparato está preparado para su uso en poco más de un segundo tras su encendido. Los 580 códigos de errores más importantes tienen un claro texto descriptivo del fallo, el cual se indica en texto deslizante (en el lenguaje seleccionado). Esta característica ayuda dando una rápida y efectiva diagnosis del problema. En la gran mayoría de los casos no será necesario buscar el código en un libro de códigos de errores OBD.

Colocación de todos los bits juntos

Se ha prestado atención al diseño de la Placa de Circuito Impreso (PCI) (**Figura 4**); no se han usado componentes SMD para este diseño para simplificar el montaje de componentes. La PCI ha sido producida siguiendo la norma industrial y usando tarjetas tipo FR4 con un chapado de oro. El oro es químicamente inerte y da a la tarjeta vida casi ilimitada. El chapado también asegura que no habrá problemas de corrosión, algo que sí se puede producir cuando se usan soldaduras sin plomo en placas sin este chapado. El oro tiene unas excelentes propiedades de conservación y permite el uso tanto de soldaduras libres de plomo como de soldaduras de plomo/estaño. Además de la necesidad de cuidar la colocación de componentes y las soldaduras, no es necesaria una especial destreza en electrónica o en el dominio de la programación para completar este proyecto.

Aparte de los dos diodos LEDs “connect”

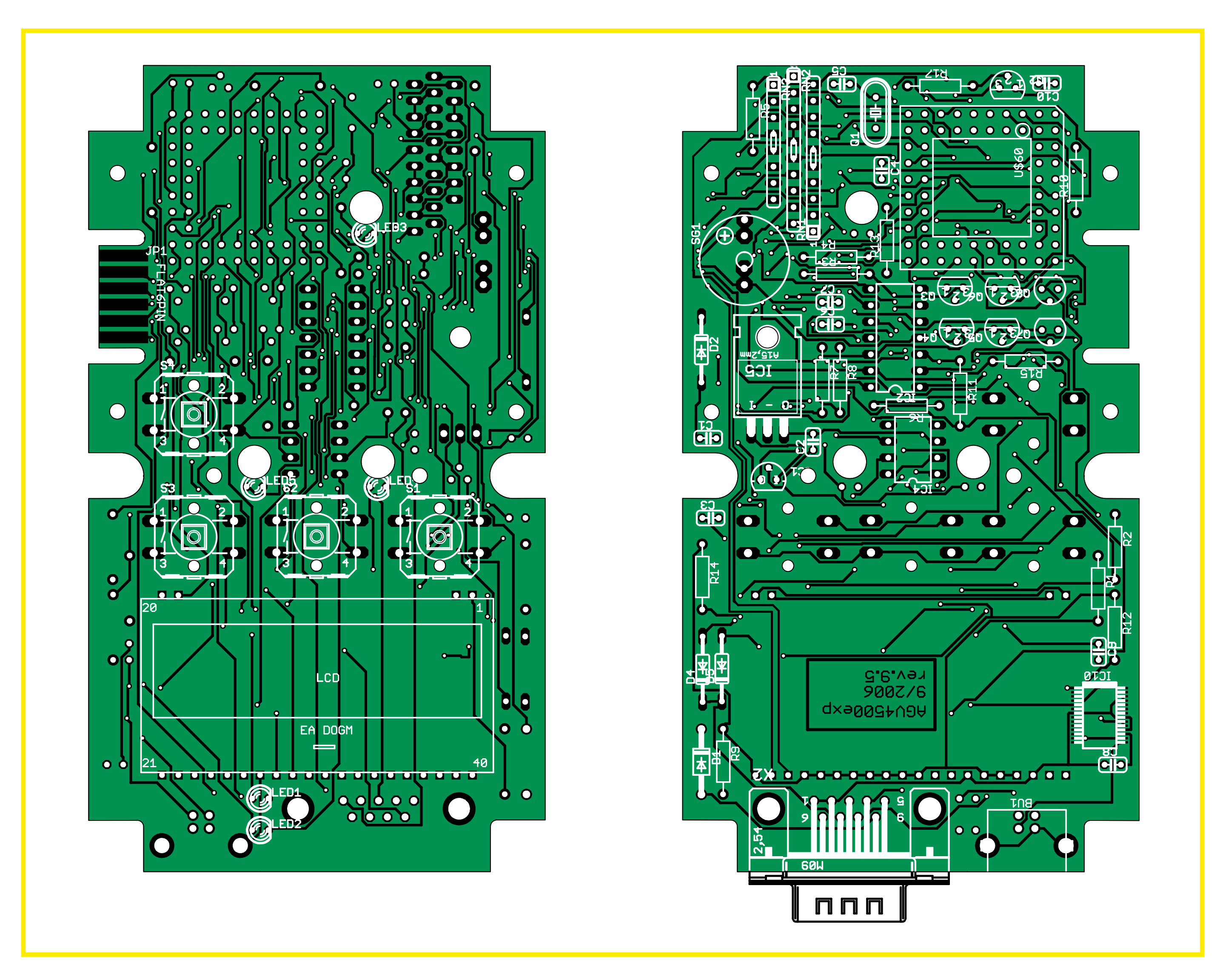


Figura 4. La Placa de Circuito Impreso (PCI) está libre de componentes SMD.

(conectado) y "data-trafic" (tráfico de datos), los dos botones pulsadores y la pantalla LCD, todos los demás componentes se montan en el lado impreso de la PCI, donde se aporta la forma de los componentes y su identificación (**Figura 5**). El montaje de componentes comienza con la soldadura de cada una de las resistencias en su lugar, seguido por los diodos, los condensadores, el cristal de cuarzo, los zócalos para los circuito integrados, las redes de resistencias (debemos asegurarnos de que los estamos montando de forma correcta), el regulador de tensión y, por último, los transistores. Antes de soldar los terminales del circuito integrado 7805 deberá fijarse primero de forma mecánica a la placa. Una vez que el timbre y el conector sub-D están fijados en la tarjeta, podemos dar la vuelta a la placa y soldar en su lugar los pulsadores, la pantalla y los diodos LED.

Por desgracia, el circuito integrado AT-

Lista de componentes

Resistors

RR1 = 1kΩ
R3,R4 = 100kΩ
R5,R9 = 33Ω
R6,R11 = 560Ω
R7,R8 = 100Ω
R12 = 1kΩ5
RN1,RN2 = 10kΩ matriz SIL de 10 resistencias
RN3 = 6kΩ8 matriz SIL de 8 resistencias

Condensadores

C1,C2,C3,C10 = 100nF
C4,C5 = 22pF
C6,C7 = 470pF

Semiconductores

D1 = 1N4004
D2,D4,D5 = 1N4148
IC1 = 78L08
IC2 = LM339N
IC3 = AT90CAN128 (de la casa Atmel, encapsulado QIL, programado como "AGV 4900").
IC4 = PCA82C250 (Philips)
IC5 = 7805
LED1 = 3mm, rojo
LED2 = 3mm, verde
Q3-Q6 = BS170 (TO92)
Q7,Q8 = BS250 (TO92)

Varios

Q1 = Cristal de cuarzo de 8 MHz (HC49/S)
LCD = Pantalla de 3 x 16 líneas, tipo EA DOGM163E; con luz de fondo: EA LED55X31-A
S1 – S3 = Pulsador para montaje en PCI tipo 40-XX B3F (Omron) con tapa de aluminio adaptada
SG1 = Zumbador DC
X2 = Conector Sub-D de 9 terminales macho para montaje en PCI
Zócalo = DIL de 14 terminales
Zócalo = DIL de 8 terminales
Zócalo = QIL (4 segmentos de 16 terminales)
PCI = Caja con lámina de serigrafía para el panel frontal.
Materiales de montaje

NOTA: el KIT de montaje N° 070038-71 contiene todos los componentes, la caja (con la lámina de serigrafía para el panel frontal montada), los materiales de montaje y el cable OBD-2. Vea la nota de publicidad en Elektor SHOP o visite la página web www.elektor-electronics.co.uk





Figura 5. Cara de componentes de la PCI.

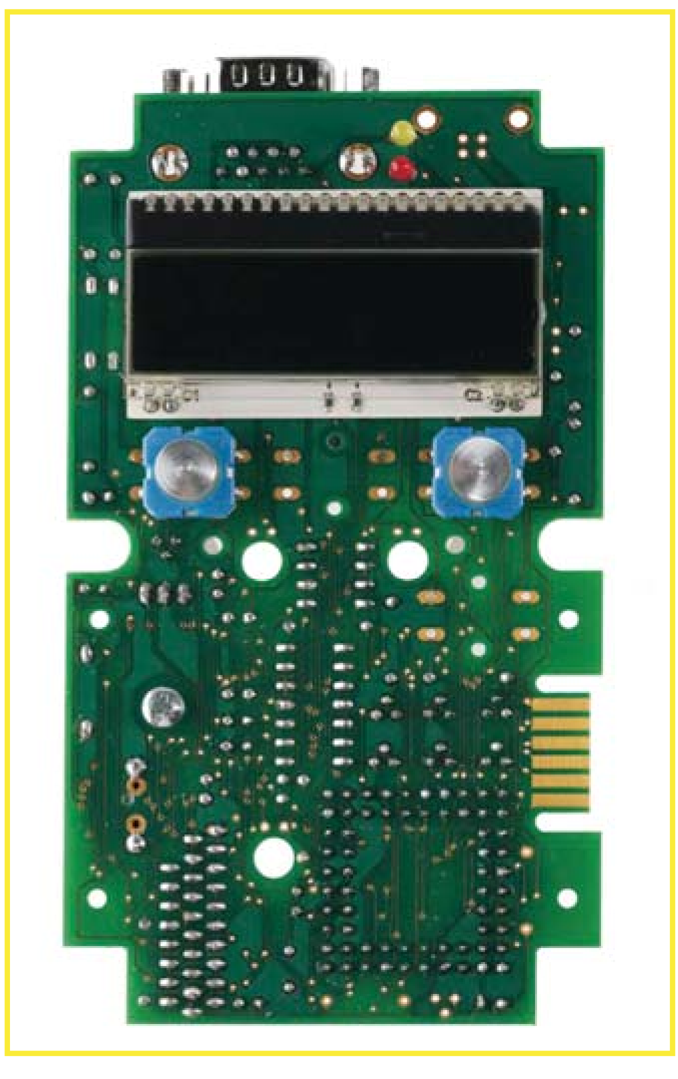


Figura 6. Los dos pulsadores, los diodos LEDs y la pantalla LCD se montan sobre la otra cara de la PCI

90CAN128 de la casa Atmel usado en este proyecto sólo está disponible en el encapsulado TQFP o en el MLF/QFN, y ninguno de estos es realmente adecuado para proyectos de auto-montaje. Por lo tanto, el controlador es suministrado (en encapsulado MLF) ya montado en una pequeña tarjeta. Sólo es necesario montar una especie de zócalo intermedio, preparado para conectar la tarjeta a la PCI principal. La distribución de terminales de este conector es el mismo que el de un encapsulado QIL64 (cuatro bloques de 64 terminales en línea). Todos los componentes para este conector están incluidos en el kit. Para asegurar el éxito del montaje será necesario seguir las instrucciones cuidadosamente, un error aquí será difícil de corregir.

El conector completo está hecho de cuatro tiras (**Figura 7**) fijadas a la PCI principal. Cada tira se fija en su lugar inicialmente soldando sólo un terminal de cada una de las tiras. Esto permite que la disposición final del conector en su conjunto sea fácilmente ajustable, hasta que cumpla exactamente al dibujo marcado en la placa. Una vez que las cuatro tiras están alineadas con precisión (verificaremos que están

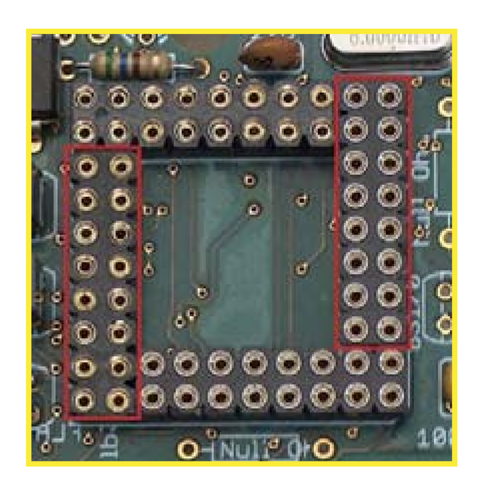
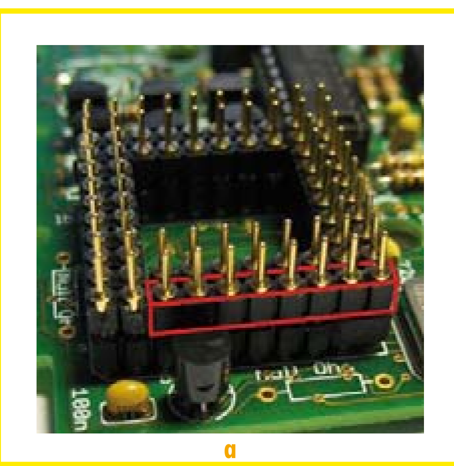


Figura 7. El conector para la placa del controlador está hecho de cuatro secciones independientes.

también todas a la misma altura de la tarjeta) todos los terminales que quedan pueden ser cuidadosamente soldados hasta completar el conector.

Vamos a montar ahora la pequeña placa que monta el controlador, que será fijada sobre unos terminales que se instalan sobre este conector. Las tiras de terminales suministradas deben separarse cuidadosamente en trozos de 8x2 terminales. Cualquier esquina rugosa puede rebajarse con una lima fina. Las tiras se introducen totalmente en el conector, tal y como podemos ver en la **Figura 8a** (éstas sólo se fijan por un lado: los terminales más finos van dentro del conector).

La tarjeta del controlador puede colocarse ahora sobre los terminales que sobresalen (**Figura 8b**), asegurándose que el pin 1 está correctamente alineado (a la izquierda de C10). Una vez montado todo correctamente, podemos ahora, cuidadosamente, soldar los 64 terminales de la tarjeta del controlador (**Figura 8c**).



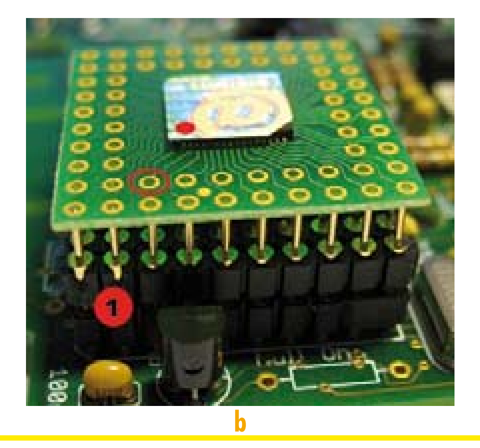




Figura 8. Tres sencillos pasos para montar la placa del controlador.

Una vez que todos los componentes se han fijado, podemos hacer un pequeño test conectando una alimentación de 12 V al conector sub-D (terminal 9 = +12 V, terminales1 ó 2 = 0 V). La corriente consumida por el analizador no debería exceder de unos 150 mA. La pantalla con iluminación de fondo se encenderá y el número de la versión de arranque cargada aparecerá en la pantalla, seguida del mensaje de bienvenida. La corta prueba inicial ya está completada.

Una vez terminada la PCI, podemos montarla en su caja: fijaremos los tapones de los pulsadores y la caja del conector sub-D, quitaremos la película de protección de la pantalla y, con la pantalla hacia abajo, colocaremos la PCI sobre la tapa frontal de la caja. Ahora podemos atornillar cuidadosamente los pequeños tornillos, dejándolos bien apretados. Por último, fijamos la otra mitad de la caja y el Analizador OBD-2- queda preparado para trabajar.

Funcionamiento del analizador

El primer requerimiento antes de que el analizador OBD-2 pueda usarse es que el coche esté dotado del correspondiente conector OBD-2 (ver el apartado "Puntos a Señalar", dentro de este artículo). Si es así, el cable OBD-2 suministrado se conecta en el conector OBD-2 del coche. El conector no debería ser demasiado difícil de colocar, la norma insiste en que debe estar montado en el vehículo, a un metro de distancia, como máximo, del asiento del conductor. En la página web www.elektor-electronics.co.uk podemos encontrar y descargar un conciso manual de operación para el analizador. Con el kit de todos los componentes también se entrega un pequeño manual, de modo que no debería haber problemas con estos puntos. En nuestra página web también hay un simulador on-line, de modo que podamos familiarizarnos "de forma virtual" con el funcionamiento y manejo del analizador.

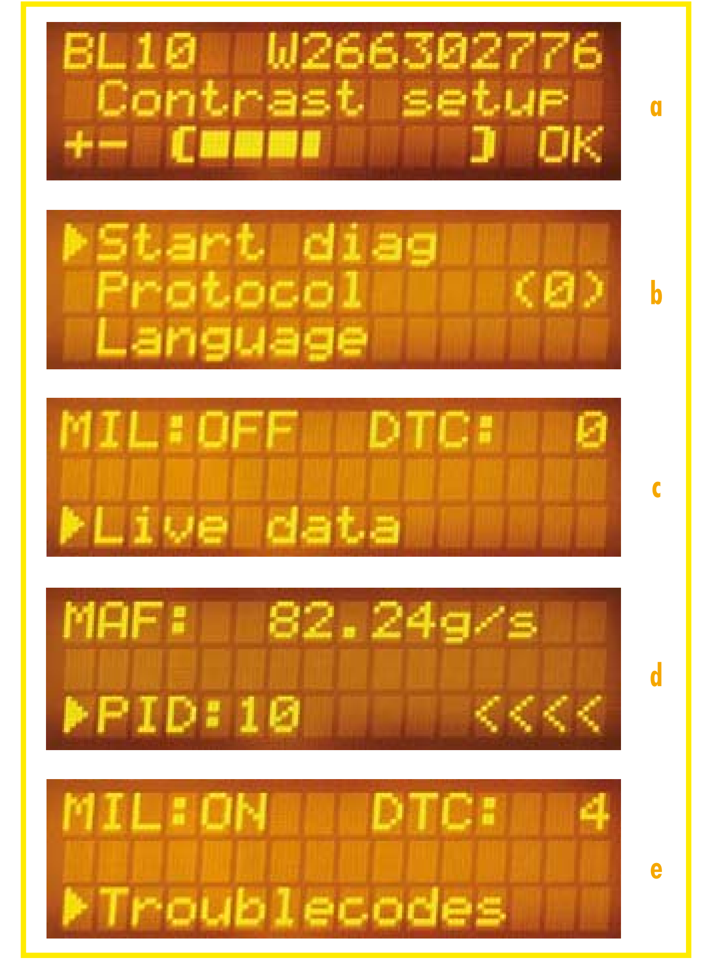


Figura 9. Presentaciones en pantalla: (a) ajuste del contraste, (b) menú inteligente, (c) presentación del estado, (d) Menú PID MIL / DTC, (e) número de código DTC de avería.

Al conectar y encender el equipo, es posible cambiar el contraste de la pantalla (**Figura 9a**). Esto se consigue manteniendo pulsadas las teclas A y B y conectando el analizador en el conector OBD-2. El contraste cambia cada vez que se presione la tecla A. Una vez que estemos conformes con el ajuste, soltamos la tecla A y presionamos la B para grabar la configuración. Este método básico se usa para controlar el analizador: con la tecla A nos movemos a través de las opciones del menú, mientras que con la tecla B confirmamos una selección o damos una respuesta desde el equipo.

La pantalla muestra ahora el mensaje de bienvenida "ELEKTOR OBD2 1.4" con el menú de arranque (**Figura 9b**) continuando, un corto periodo de tiempo después, con las opciones: Start Diag (Inicio del diagnóstico), Protocol (Protocolo) y Language (Idioma). Cuando el analizador se usa regularmente en el mismo vehículo y estamos seguros del protocolo adecuado, podemos seleccionar la opción "código 0" para que el analizador encuentre automáticamente el protocolo correcto. Presionando la tecla B comienza la búsqueda (si la interfaz del vehículo no es compatible con el protocolo OBD el test finaliza con un mensaje de fallo). Cuando el proceso de búsqueda ha finalizado, la

DTC:2/4 P0230
Fuel Pump Prim
▶Next code

errores encontrados: 2 de 4
código de error 2
descripción del error 2
selecciona el error 3 con el botón B

070038 - 13

Figura 10. Menú con el código de avería que muestra una descripción de la misma en un texto que se desplaza.



Figura 11. Menú con la trama capturada (PID seleccionado).

pantalla mostrará el estado del MIL/DTC, indicando si hay algún problema almacenado en memoria (**Figura 9c**). Seleccionando la opción Live Data con el botón B se mostrará el valor actual del parámetro. El número de chasis o el Identificador (ID) del vehículo puede leerse y se mostrará el protocolo de comunicaciones, así como la opción volver a buscar.

En la opción de menú PID (parámetro identificador) se da la lectura actual de un sensor (datos en vivo). El ejemplo mostrado en la **Figura 9d** es una lectura del consumo de Flujo de Masa de Aire (MAF en g/s). Presionando la tecla B volvemos al menú anterior. Cuando se detecta un fallo por el sistema de gestión del motor, el analizador indicará que el MIL está activado (MIL:ON), y se proporciona el número de DTCs almacenado (Disgnostic Trouble Code, es decir, Código de Fallo de Diagnóstico) (**Figura 9e**). Ahora podemos elegir entre presentar los códigos de fallos o las tramas de datos capturadas. Para los códigos de fallo, el número de código se indica (en la mayoría de los casos) con una descripción detallada del fallo (**Figura 10**).

Cuando se muestra el código del fallo, al presionar la tecla A aparece una opción para borrar los códigos de la memoria del vehículo.

Podemos obtener más información acerca del fallo seleccionando “Freeze Frame” (capturar trama). Cuando el sistema de gestión del motor detecta un fallo, el ordenador de a bordo realizará una captura de una imagen o “congelará la trama” de todos los sensores leídos y almacenará todo en la memoria del vehículo. Un chequeo de este dato puede dar una valoración de la causa del fallo. Al seleccionar los códigos de fallo de la trama capturada, es posible seleccionar sucesivos valores de sensores, almacenados en torno al momento en que se produjo el fallo.

El ejemplo de la **Figura 11** indica la lectura de un sensor cuando se registró el fallo; fallo F000, un PID 0D (velocidad) medido de VSS = 33 km/h. El botón A nos lleva a través de sucesivos parámetros mientras el botón B vuelve al menú anterior.



El manual de usuario, que se puede descargar de Internet, contiene vistas generales de todas las opciones del menú, selecciones y mensajes de pantalla. Cuando queramos una mayor familiaridad con el funcionamiento del equipo, podemos probar el simulador online mencionado anteriormente, o mejor aún, ¡haz tu pedido y construye tu propio analizador OBD-2!. También podemos descargar un fichero extra que contiene consejos de soldadura y ensamblaje, desde la página web de Elektor.

(070038-I)

Literatura

[1] Datasheet for the AGV4900-Controller: www.obd-diag.de

[2] Source of the AGV microcontroller: www.stange-distribution.de

[3] http://www.blafusel.de/misc/OBD-2_scanned.php



Inspector Digital

Analizador lógico de cuatro canales

Ronald de Bruijn

El analizador lógico es indispensable para comprobar las señales digitales, especialmente hoy en día que cada vez más circuitos están dirigidos por microconntroladores. En este artículo presentamos un circuito fácil de montar que sabe como manejar perfectamente compatible con la mayoría de señales digitales a la perfección y además dispone de una función de memoria.

Especificaciones

Frecuencia de muestra:	200 Hz a 2 MHz
Número de canales:	4
Rango de medición:	de 0 a 5 V
Memoria:	1024 muestras por canal
Estado de activación:	+ y –
Patrones de activación:	a establecer para cada entrada
LCD gráfico:	64 x 128 píxels
Alimentación:	batería de 9 V

Para controlar comprobar las señales digitales, se utiliza un analizador lógico. A veces es práctico poder hacerlo in situ, sino o si debe hacerse una 'medición flotante'. El analizador lógico de 4 canales que presentamos a continuación es ideal para eso ello gracias a su compacidad y la alimentación por pilas. La velocidad de muestra muestreo asciende como máximo a 2 MHz y el circuito dispone de suficiente memoria para almacenar temporalmente 1024 muestras de la señal. La pantalla gráfica de 64 x 128 píxels reproduce claramente las señales lógicas.

Esquema

La pieza principal del circuito es el IC2 (un PIC18F4850, véase **figura 1**). Este controlador PIC examina las señales y gestiona el visualizador. Se maneja con cinco botones pulsadores (S1... S5). El cristal (X1) de 10 MHz determina en gran medida la frecuencia de muestraeo más elevada posible. El PLL interno del microcontrolador está conectado-se utiliza para entregar al controlador una frecuencia de reloj interno de 40 Mhz, con lo cual el controlador funciona con una velocidad del reloj interna de 40 MHz, la velocidad máxima que recomienda Microchip para este tipo de integrados.

En la entrada, los D1 a D8 protegen contra tensiones negativas o demasiado altas o demasiado bajas. Las señales de entrada pasan a través del IC1, un 74HC04N que se utiliza como buffer. Si por elloDe echo la señal se invierte, pero no importa en este caso. Lo podemos corregir con el software. Luego, las señales entran directamente en el controlador por RA1...RA4, y luegodonde el software se ocupa de ello ellas (zie *Bediening*).

Con P1 podemos establecer el contraste de la pantalla y con T1 encender la su iluminación de fondo de la pantalla. BZ1 da un aviso acústico cuando se inicia un nuevo ciclo de muestra y cuando se conmuta del modo run al modo hold.

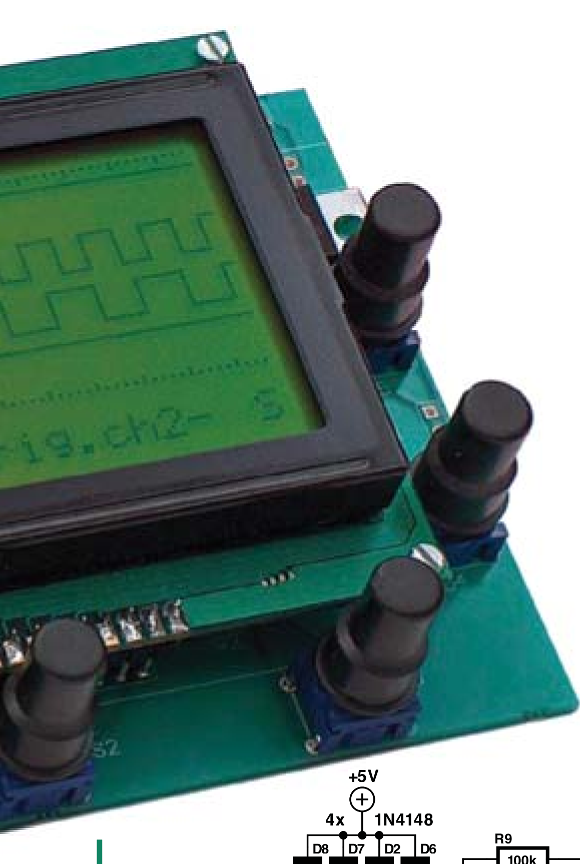
Los cinco interruptores utilizados para el manejo control no requieren circuitería de eliminación de vibraciones rebotes, ya que el software se encarga de ello. a través del hardware. Se hace con el software.

La alimentación del circuito consta de dos partes: una tensión estabilizada a 5 V y una tensión de 9 V para la iluminación de la pantalladel visualizador. La fuente de alimentaciónde que suministre estas tensiones puede ser un adaptador de red de 9 a 12 V o una batería de 9 V.

También dispone de un sencillo circuito de carga simple integrado para la batería (T2, R1, R17, D12) que se pone en marchaconecta en cuanto se conecta un adaptador de red. Partiendo de un LED estándar con una tensión directa Vf de 1,5 V, la batería se carga con:

(1,5-0,6)/56 = 16 mA.

Así, una batería de NiMH de 9V de 170 mA se carga aproximadamen-

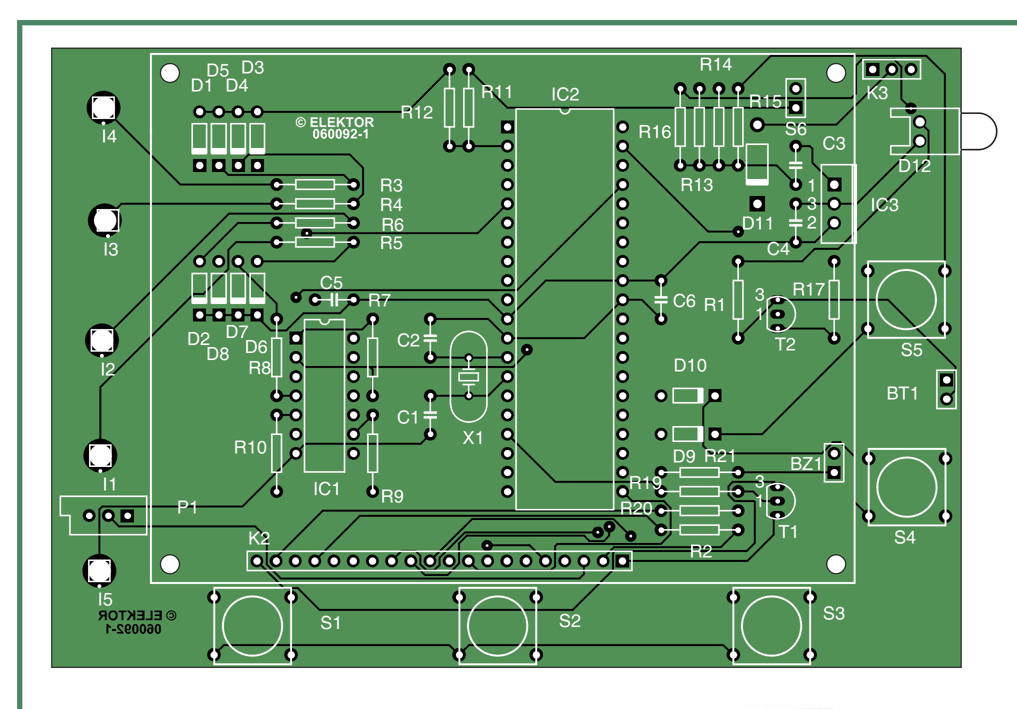


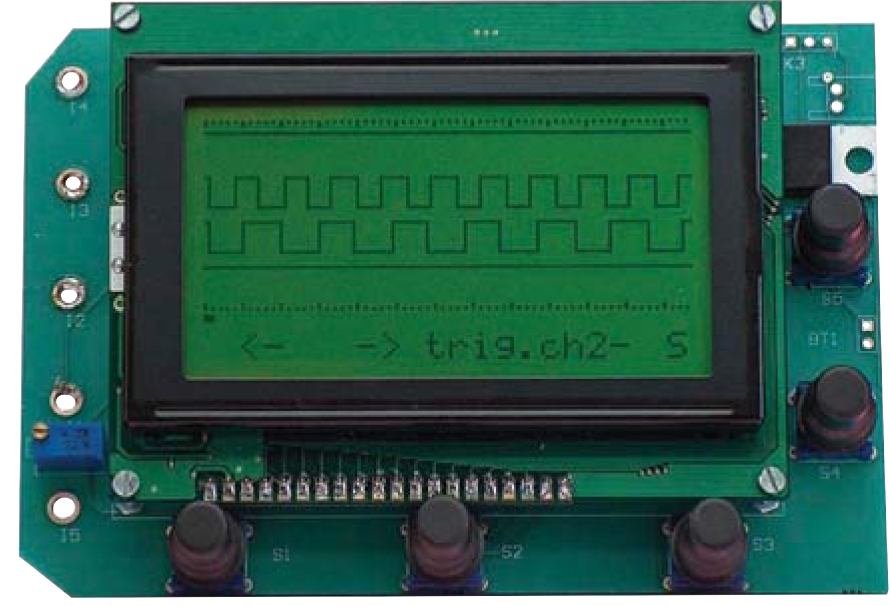
te con 1/10 C, de modo que la carga continua no cause daños. La batería se carga en unas 10 horas con esta conexión. Durante la carga permanece encendido el LED D12. Si se utiliza una pila normal (no recargable), el circuito puede caer cerca de T2.la circuitería entorno a T2 puede eliminarse.

Manejocontrol

Con el interruptor S1 se elige la frecuencia de muestramuestreo. Las velocidades que deben ajustarsese pueden eleguir son 5/10/20/50/100/200/500 μs/div y 1/2/5 ms/div. S2 sirve para seleccionar el canal en el que se activa el circuito. Con S3 se puede establecer si el PIC debe activarse en flanco ascendente o descendente y con el S4 se activa o detiene el circuito, o se vacía borra el contenido del visualizador. Pulsando una vez sobre S4 se arma el circuito. Tras recibir la señal de disparo, se leen y almacenan 1024 muestras por canal tras una señal de activación. Si se pulsa de nuevo brevemente S4 el

Figura 1. En el esquema se puede ver claramente que el microcontrolador se encarga de todo.





Lista de componentes

Resistencias:
R1 = 680 Ω
R2,R11...R16 = 10 k
R3...R6 = 330 Ω
R7...R10 = 100 k
R17 = 56 Ω
R19 = 1 k
R20 = 47 Ω
R21 = 220 Ω
P1 = 20 k potenciómetro de ajuste, modelo multiturn, vertical.

Condensadores:
C1,C2 = 22 p
C3...C6 = 100 n

Semiconductores:
D1...D10 = 1N4148
D11 = 1N4001
D12 = 5 mm LED
T1,T2 = BC337
IC1 = 74HC04
IC2 = PIC18F4580-I/P (programado, EPS 060092-41)
IC3 = 7805

Varios:
Bz1 = Buzzer AC
X1 = cristal 10 MHz
Visualizador LC gráfico 128 x 64 píxels, pej DEM128064A o NLC128x64 (núm. pedido Conrad 187429)
Caja 186 x 123 x 41 mm con compartimiento para batería de 9 V, pej. Strapubox (núm. pedido Conrad 522775)
S1...S5 = botón pulsador Multimec RA3FTL6 con botón AQC09-24.2
S6 = interruptor unipolar
Clip de batería de 9 V
5 conectores tipo banana para montaje con tornillos (para conexión con I1...I5)
Paquete completo con caja: EPS 060092-71
Diseño de placa descargable gratuitamente en la web, 060092-1

Figura 2. Como se puede ver en la instalación de los componentes, la estructura del circuito no es complicada. El conector K2 se coloca de modo que el visualizador se pueda colocar directamente sobre la placa de doble cara.

circuito lee 1024 muestras nuevas tras recibir unala siguiente señal de activación. Si mantenemos pulsado S4 de forma prolongada, se borra la pantalla. Entonces se almacenan los últimos ajustes de la frecuencia de muestra-muestreo, el canal en el que se ha activado y el estado de activacióny condiciones de disparo en ella EEPROM del microcontradormicrocontrolador. Estos ajustes se usan como valor de inicio al encender de nuevo el circuito.
Con S5, se enciende o apaga la iluminación de fondo del visualizadorde la pantalla. Tras un minuto y medio aproximadamente o en caso de 'Lo_ Batt' (bateria baja) el microcontrolador apaga automáticamente la iluminación de fondo.

Funcionamiento

Para obtener una velocidad de muestra muestreo lo más elevada posible, dejamos que el microcontrolador, tras una señal de activación, cargue primero las muestras en su memoria RAM. Esto lo hacemos con la instrucción de software *movff port a, postinc0*. Esta instrucción copia el contenido del puerto a en RAM y luego incrementa a uno la dirección RAM. Este ciclo transcurre como máximose repite 1024 veces. Al terminar, se han leído y mostrado en el visualizador 128 muestras de la memoria RAM. Este proceso se repite cada 1 s.
Si durante 3 s (según dependiendo de la frecuencia de muestra) no se ha producido ningúnninguna nueva condición de disparo estado de activación, el circuito lee 128 muestras y las muestra presenta en el LCD. Así se puede ver en qué estado (alta o baja) se encuentran las entradas.
Una pulsación corta en S4 activa la función de memoria. Como indicación aparece de nuevo en la pantalla una

'R'Ello queda indicado a la derecha de la pantalla con una 'R'. El circuito espera ahora una señal de activacióndisparo. Cuando ocurre, y se almacenan 1024 muestras, la 'R' cambia a 'S' y el visualizador muestra las primeras 128 muestras de cada canal. Con S1 y S2 se puede ir pasando por las posiciones de memoria. Pulsando brevemente en S1 o S2, el visualizador avanza poco a poco por la memoria; Pulsando más tiempo en S1 o S2 la pantalla avanza más rápido. El cursor en la parte inferior de la pantalla muestra se indica qué parte de la serie de muestras se está reproduciendomostrando en cada momento.
Pulsando de nuevo brevemente-Una pulsación rápida sobre S4 hace que, el circuito lee lea de nuevo los valores de entrada y los fija almacena en la memoria. La pantalla sigue mostrando la misma zona de memoria que se estableció primeropara las muestras anteriores. Evidentemente, esto es muy práctico para estudiar las señales que siguen a una señal de activacióndisparo.
Si se pulsa más tiempo S4, suficiente para que el zumbador dé una señal, el circuito leerá de nuevo sus entradas y almacenará los valores medidos en su memoriaun nuevo conjunto de muestras y los almacenará en la memoria. Ahora, sin embargo, no se muestra la misma zona de memoria, sino que el visualizador salta al principio de la memoria.
Si se pulsa aún más S4 (hasta que suenen dos señales acústicas, una tras otra), el analizador lógico sale del modo memoria y regresa a su modo estándar en el que se muestran presentan siempre 128 muestras 'live' (en directo) en el visualizador.

Construcción

Para este diseño no se utilizan SMD. La estructura placa es bastante amplia y todos los componentes son accesiblesfácilmente localizables. Por lo tanto, al soldar no habrá problemas.
Sin embargo, quisiéramos comentar la conexión entre el visualizador y la impresoraplaca. El visualizador puede colocar sobre la impresoraplaca. El método más práctico para conectar el visualizador a la impresoraplaca, es soldar primero un pinheader simple en la impresora del visualizadoruna tira de pines a la placa de la pantalla. Luego se inserta un enchufe de alambre enrollado en el pinheader y se coloca todo en la impresora principal.Después, enchufamos en esa tira de pines un zócalo de grapinado y lo insertamos en la placa principal Debe comprobarse que el visualizadorla pantalla se encuentra a la altura adecuada y después se suelda el enchufe de alambre enrollado en la impresora principal.zócalo de grapinado a la placa principal.
Si se integra el circuito en la caja recomendada, deben limarse las dos esquinas de la impresora del lado de las entradas de señal. Entonces la impresora entrará como si estuviera moldeada.Cuando se utilice la caja recomendada para este circuito, hay que limar primero las esquinas de placa por la parte de las señales de entrada. Entonces la placa encajará perfectamente.

Observaciones

Si no se usan todos los canales, es recomendable conectar a masa los no utilizados. Si no se (desea) conectan todos los canales al mismo tiempo, entonces se recomienda conectar los canales no usados a tierra. En muchos casos las entradas abiertas pueden captar interferencias, que reproducen todo tipo de señales inútiles en el visualizador.
Es muy posible que este analizador no seaDebe quedar claro que este analizador no es apto para frecuencias muy altas. Por lo tanto, las aplicaciones de este aparato de medicióndispositivo deben elegirse parason los microcontroladores "lentos", control de buses seriales lentoscomunicaciones serie, etc. Sin embargo, este sencillo circuito simple ofrece es muy útil para prototipos de un diseño digital (microcontrolador).puede hacer nuestra vida más fácil durante el desarrollo de un circuito digital (microcontrolador).
De Para este diseño, ofrecemos un paquete de montaje (EPS 060092-71). Este paquete contiene el visualizadorla pantalla, la impresoraplaca, el microntrolador microcontrolador programado, las piezaslos componentes y la caja. Lo único que hay que hacer, es soldar los componentes en la impresora placa y montarlo todo en la caja. Una vez conectada la batería o la alimentación, se puede empezar a medir analizar inmediatamente.
Para los que prefieran programar confeccionar ellos mismos la impresoraplaca, se pueden descargar el diseño de la impresorade la misma de nuestro sitio web, www.elektuur.nl, bajo el número EPS 060092-1. Si desean programar el microcontrolador PIC, también pueden obtener el código fuente en nuestro sitio (EPS 060092-11).

(060092)

Bajo coste – Baja potencia
Servidor web AVR

del ingeniero (Escuela de ciencias) Holger Buss y Ulrich Radig, en colaboración con el Dr. Thomas Scherer

Un servidor web con controlador Atmel. ¿No puede ser? ¡Pues claro que sí! Al parecer en el mundo del código abierto nada es imposible, así que se ha extraído el código necesario para un ATmega3. Incluso se pueden controlar elementos externos y conectar una cámara web.

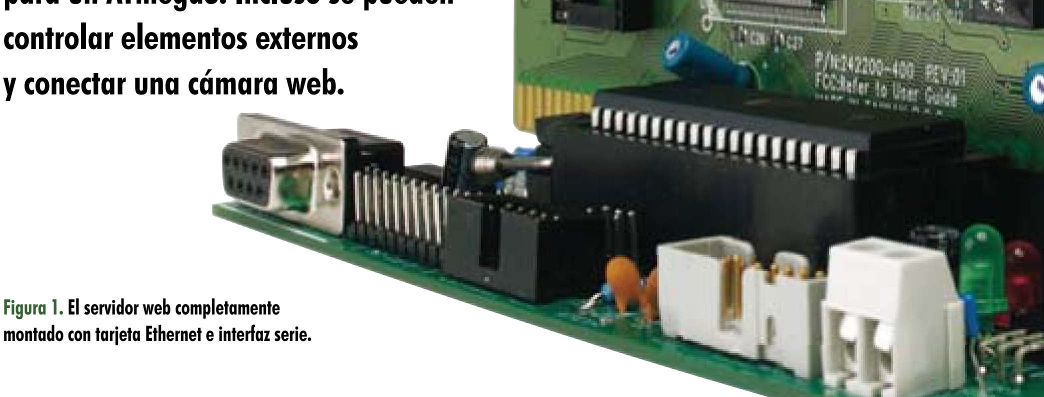


Figura 1. El servidor web completamente montado con tarjeta Ethernet e interfaz serie.

En realidad al principio Ulrich Radig sólo quería encender su máquina de café por internet. Tras algunas sutilezas, de ahí surgió un pequeño servidor web en el que se podían controlar algunos puertos y escoger entradas análogicas.

Después de haber presentado su proyecto en internet [1], enseguida encontró un aliado que trabajaba con hardware y software. Holger Buss desarrolló una placa que establecía la conexión con internet gracias a una vieja tarjeta Ethernet compatible con NE2000 [2] (**Figura 1**). A diferencia de muchos otros pequeños proyectos de servidores, este circuito funciona completamente sin SMD, así que es adecuado para montarlo por cuenta propia, y además no sólo recicla electrónica antigua, sino que es muy barato de implementar.

Características de rendimiento

- Servidor web con ATmega32/ATmega644
- Consumo de corriente <100 mA
- Consulta de tres entradas analógicas
- 7 I/O digitales
- 1 interfaz de cable (Dallas)
- Conexión para cámara web
- Interfaz para tarjeta de memoria SD
- Consulta de un servicio de sincronización de tiempo NTP
- Configuración mediante archivos de texto
- Aviso por correo electrónico
- Interfaz en serie
- Programación dentro del circuito
- Ethernet mediante una tarjeta de red ISA compatible con NE2000

¿Todo Linux o qué?

Normalmente uno se imagina un servidor web como un ordenador ruidoso en una carcasa de 19" – *diseñado para un centro de procesamiento de datos* – y con Linux o Windows como sistema operativo. Hoy en día esa imagen está cambiando, y uno ya está acostumbrado a formatos más pequeños, ya que algunas carcasas para discos duros externos no sólo incluyen interfaces de USB, SATA o Firewire, sino también con las tres letras NAS (**N**etwork **A**ttached **S**torage). Por lo general contiene un controlador rápido con una velocidad de unos cientos de MHz, en el que funciona un Linux reducido. Se pueden configurar las propiedades del servidor para el protocolo FTP, SMB y a veces también el HTTP a través de una página web incorporada. Son necesarios muchos MB de memoria flash y RAM. Pero un ATmega32 sólo tiene 32 **Kilo**byte de memoria Flash, y en la RAM hay que contentarse con 2048 bits (!). Además, un microcontrolador así funciona a una velocidad de 16 MHz con sólo una fracción de la velocidad de un controlador NAS. Y, no obstante, el "32" del nombre de la serie ATmega32 se refiere a puras CPUs de 8

bits. Eso recuerda mucho a los principios de los microordenadores hace ya 25 años, cuando se podía comprar por mucho dinero algo parecido en la forma a una placa grande, casi del tamaño DIN-A3, con CPUs Z80, 5602 o 6800. Hoy en día se puede obtener un ATmega32, claramente con un mayor rendimiento, por menos de 5 €, incluso la variante Atmega644, con el doble de memoria flash, por menos de 10 € en la tienda. Este minimalismo económico y técnico tiene una consecuencia clara: no es Linux.

¿Código abierto?

Así pues, no queda nada más que comprimir los componentes absolutamente imprescindibles del protocolo de internet, que se ajusten a la memoria flash de un controlador AVR. Los autores no estaban solos con semejantes exigencias, ya que ya existían trabajos preparatorios de muchos otros programadores, y lo bonito del código abierto es precisamente que uno no tiene que inventar de cero cada componente. De todos modos, en no tardaron mucho en realizar un servidor web sobre la base de un controlador AVR. Existen no sólo los códigos fuente completos, sino que también están disponibles los archivos hex preparados y compilados en la página web del proyecto [3] y [4], con los que también los que estén menos familiarizados con el software puedan grabar en la memoria flash con comodidad, lo que por otra parte reduce errores. Incluso existe un foro [5] extra si en algún momento se necesita ayuda.

Lo que el pequeño servidor “puede hacer“ se encuentra en la carcasa con los datos de potencia. Es asombroso de lo que es capaz esta pequeña electrónica. Además de entradas y salidas, es especialmente interesante la inclusión de una económica tarjeta de

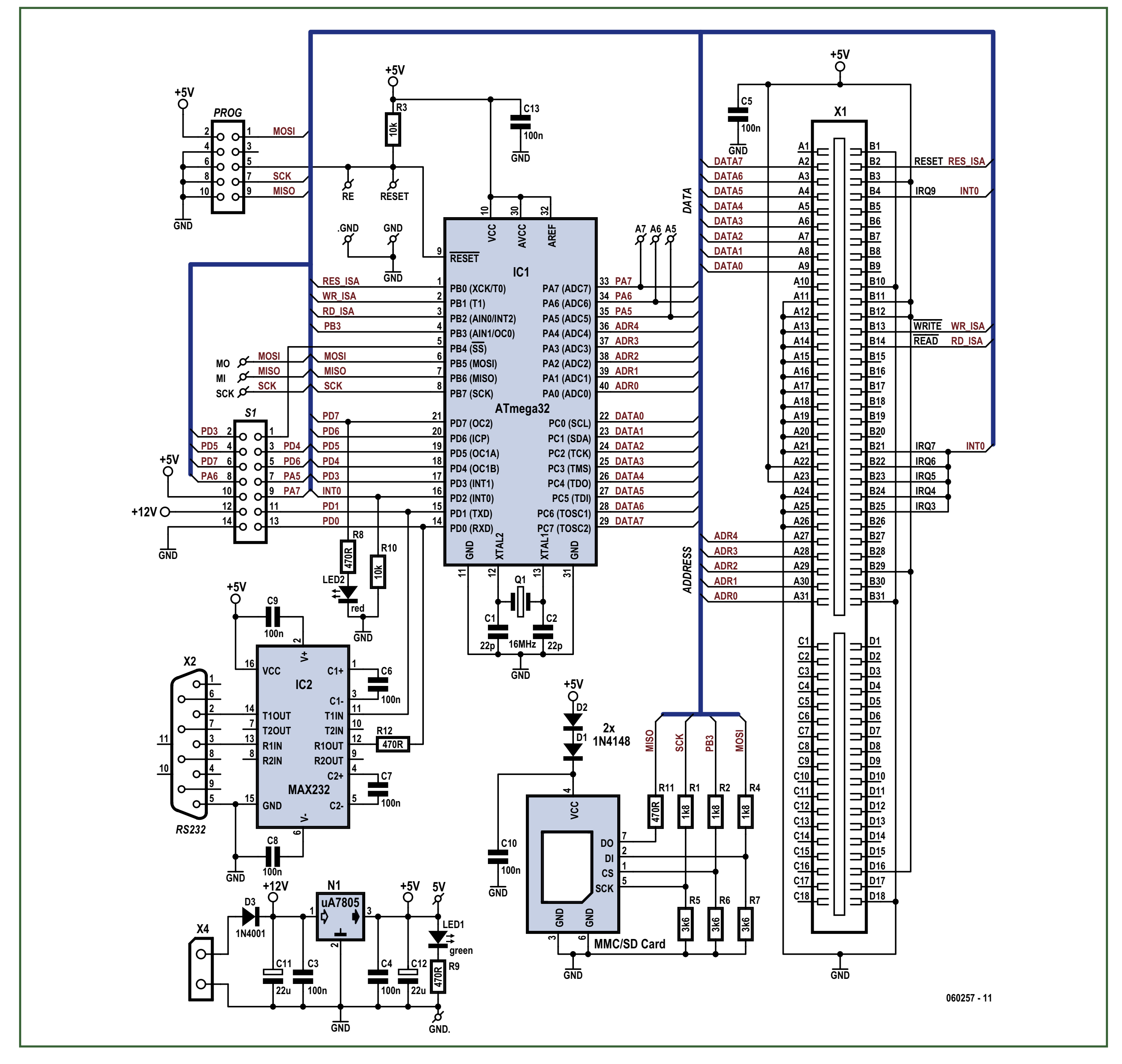


Figura 2. **El circuito del servidor web es sorprendentemente sencillo, ya que la conexión en red se traslada a una tarjeta Ethernet.**

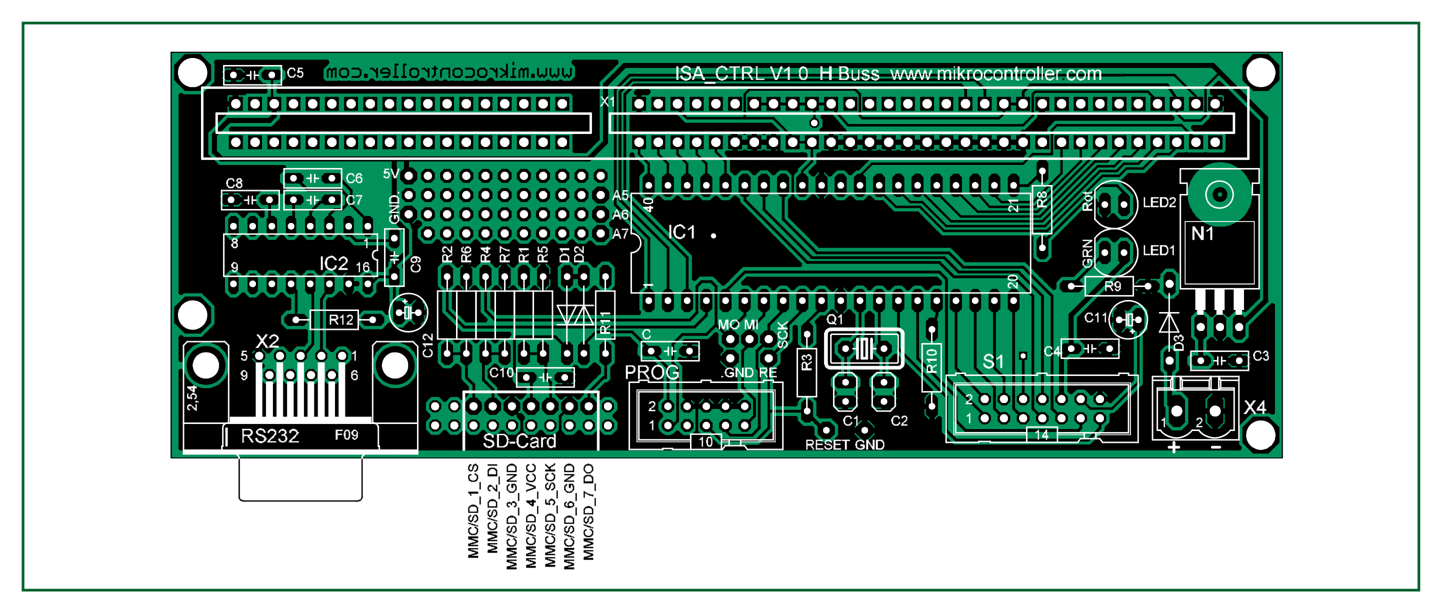


Figura 3. La placa del servidor web.

memoria SD. Así, se pueden optimizar fácilmente los parámetros de configuración o modificar la página web que debe mostrar el servidor. Y un sistema tan minimalista es muy seguro frente a los ataques de los hacker.
Donde hay luz, también hay sombras. El pequeño servidor no está pensado como sustituto de un servidor grande "de verdad". Recibe demasiada poca cantidad de datos (se consigue descargar cosas a unos 10 kB/s) y, aparte de HTTP, no sería fácil soportar otros pequeños detalles como como scripts, etc. El pequeño servidor es fantástico como control o sensor remoto de temperaturas, o para hacer fotos con una cámara web. Su consumo de energía, menos de un vatio, seguro que tampoco es ecológicamente perjudicial.

ATmega y Firmware

En el circuito de (**Figura 2**) domina el microcontrolador. Además de este circuito integrado, también está el convertidor de nivel MAX232 para la interfaz en serie y un regulador de tensión de 5 voltios. No se utiliza ninguna comunicación en serie y no se necesitan 12 voltios, así que se puede alimentar el servidor web directamente con 5 V. Por lo tanto, se eliminan IC2, N1 y D3 y en su lugar hay un D3 conectado en puente, así como un puente de alambre entre los terminales 1 y 3 del regulador de tensión. Así, el resultado es – aparte de la tarjeta Ethernet – un servidor web de un IC que sería difícil imaginar más pequeño. El consumo de energía debería estar por debajo de los 500 mW.
En las páginas web de los autores hay algunas versiones del firmware para el que es suficiente la memoria flash de un ATmega32. Sin embargo, para la versión más actualizada, hasta el momento, la 1.4, se necesita un poco más de espacio, de manera que para esto se necesita el tipo ATmega64. ¡Los más astutos incluso han reducido, por ambición sana, el código fuente hasta tal punto que, si se renuncia al lujo, encaja en un ATmega8! Encuanto a la electrónica, no es muy importante si se utiliza un ATmega16, ATmega32 o ATmega644 en un encapsulado DIL-40, ya que son compatibles en sus pines.
El firmware de 1.4 se compila para una frecuencia de cuarzo de 14,7456 MHz, porque así se puede transferir fácilmente la velocidad en baudios correspondiente para la interfaz en serie. En el caso del ATmega644 se regalan algunos MHz porque este chip resiste sin problemas 20 MHz. Si alguien desea juguetear con una mayor velocidad y sacar un poco más de rendimiento al hardware, tiene que adaptar el código fuente o renunciar a la comunicación en serie.
La tarjeta SD en realidad necesita 3,3 V para funcionar. Con los dos diodos D1 y D2 se produce una tensión de funcionamiento de unos 3,5 V. Como las tarjetas SD no son críticas respecto de la alimentación, es suficiente.
Además de las interfaces en serie, también hay dos interfaces SPI para la programación dentro del circuito con programadores propios (p.ej. el USBprog del número de octubre de ELEKTOR) con 10 o 6 pines. La tensión de alimen-

Lista de materiales

Resistencias:
R1, R2, R4 = 1k8
R3, R10 = 10k
R5...R7 = 3k6
R8, R9, R11, R12 = 470Ω

Condensadores:
C1, C2 = 22p, de cerámica
C3...C10, C13 = 100n lámina, 5 mm retícula
C11, C12 = 22µ/16V, tantalio

Semiconductor:
IC1* = ATmega32 o ATmega644, DIL 40
ATmega644 programado: 060257-41 (Tienda Elektor)
IC2 = MASX232
N1 = 7805, regulador de tensión
D1, D2 = 1N4148
D3 = 1N4001
LED1 = LED, verde
LED1 = LED, rojo

Varios:
Q1* = cuarzo, 14,7456 MHz
S1 = Tira de pines DIL de 2x7 contactos
X1* = ranura de 2x31 contactos (Reichelt) o ranura ISA
X2 = enchufe sub D de 9 pines para el montaje de la platina
Prog = tira de pines DIL de 2x5 contactos + tira de pines de 2x3 contactos
SD* = tira de pines DIL de 2x7 pines
1 zócalo para circuito integrado de 40 pines
1 zócalo para circuito integrado de 16 pines
Placa de circuito impreso 060257-1 (Tienda Elektor)

* = véase el texto

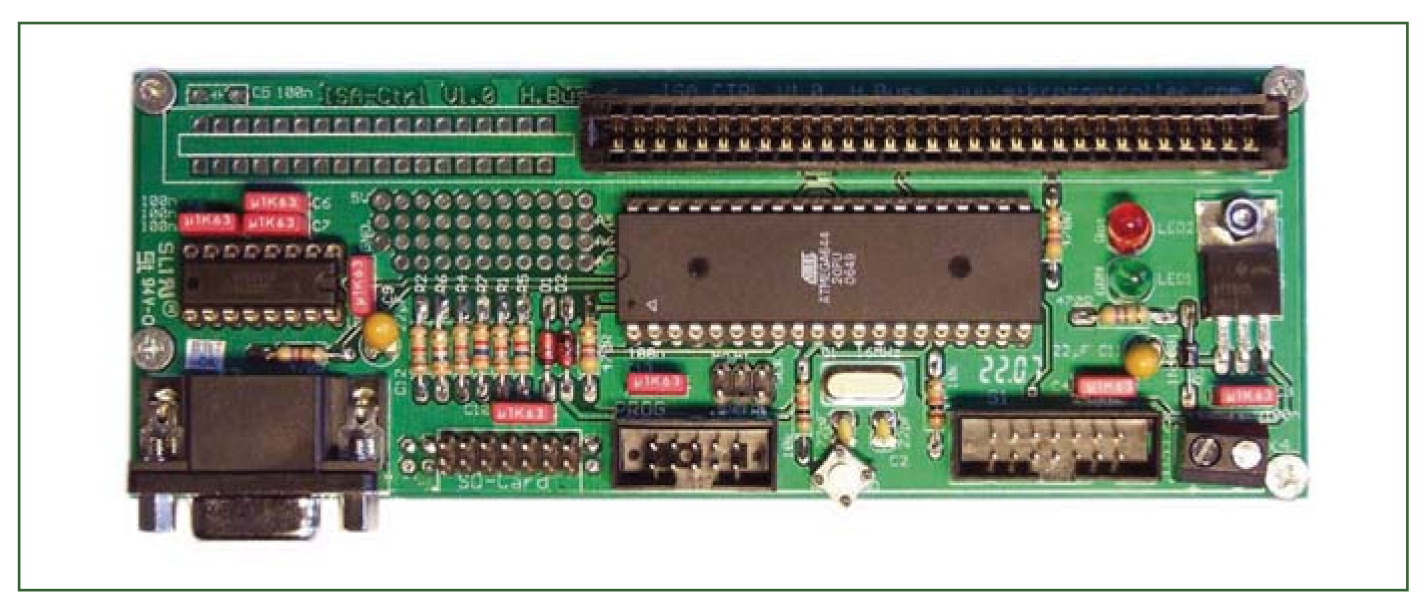


Figura 4. El servidor web con los componentes montadas. Entre los dos queda un hueco libre para hacer ampliaciones.

tación y los puertos I/O están accesibles en la regleta de 14 contactos S1, a la que se pueden conectar mediante cables los periféricos. Para incorporar la tarjeta Ethernet está prevista una clásica ranura ISA.

Ethernet

Como ya se ha mencionado, una tarjeta de la época de los PCs "XT" y "AT" sale ganando. En aquella época, anterior al PCI, las llamadas tarjetas Ethernet de 10 Mbits compatibles con NE2000 para el bus ISA eran casi un estándar. Estas tarjetas funcionan simplemente con 8 bits, ideal para un microcontrolador de 8 bits. Funcionan especialmente bien las tarjetas con un controlador de la marca Realtek. Al revolver en eBay y otras bolsas de artículos de ocasión, hay que fijarse en las tarjetas con el chip "RTL8019AS". Éstas funcionan sin modificaciones. Sólo hay que quitar una EPROM que incluye a veces, ya que puede contener software que perjudique el Netboot, etc.
Como ahora estas tarjetas ya no son tan fáciles de encontrar, hay una modificación [7] para utilizar tarjetas 3Com con un chip 3C5x9. Evidentemente, entonces hay que adaptar también el código fuente.
Por otra parte, no es imprescindible buscar una ranura ISA original. Las tarjetas también funcionan en una ranura de 2x31 polos, que están disponibles a buen precio en tiendas de electrónica como Reichelt. De todos modos, los contactos de 2x18 de la placa (**Figura 3**) que faltan sólo están unidos a masa y +5 V. Las dos líneas están prácticamente siempre conectadas en puente con la tarjeta Ethernet. También la placa ya montada de la **Figura 4** sólo muestra la ranura de 2x31 contactos.

Piezas, montaje y comprobación

Como no entra en acción ni un solo SMD, el montaje de la placa es muy fácil. Para IC1 y IC2 son recomendables zócalos. En IC1 sobre todo, ya que habrá que quitar el microcontrolador de su zócalo cuando se grabar su memoria flash con un programador, como por ejemplo el STK500 de Atmel, o se tiene intención de realizar más modificaciones ocasionales.
Está prevista como zócalos para la tarjeta SD un tira de 2x7 pines, que se suelda en la posición indicada en la placa. No es tan fácil conseguir los zócalos adecuados para tarjetas SD que se puedan soldar, y cuando se encuentran tienen un precio desorbitado.
Hay 2,5 alternativas:
La "media alternativa" se trata simplemente de un truco. Como se muestra en la **Figura 5**, las dos filas de pines se doblan entre ellas de manera que en los extremos están separadas aproximadamente a 1,5 mm. En el medio se puede fijar simplemente una tarjeta SD. Entonces adquiere el aspecto que se muestra en la **Figura 6** y es bastante seguro para experimentos.
Una posibilidad perfectamente válida consiste en utilizar una tarjeta SD mini o micro con un adaptador para el tamaño estándar, y soldar el adaptador directamente a los siete pines interiores

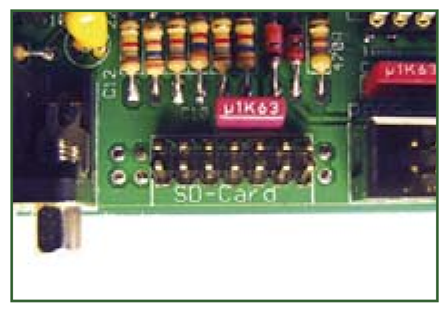


Figura 5. Para poder conectar una tarjeta de memoria SD entre las los pines del conector, se doblan las dos filas entre sí hasta quedar a una distancia de aproximadamente 1,5 mm.

(los siete pines del borde de la placa no tienen ninguna función eléctrica).
Más elegante, y aun así económico, es el sacrificio de un lector de tarjetas SD barato como en la **Figura 7**. Se extrae el "hueco" de la tarjeta SD y, como se

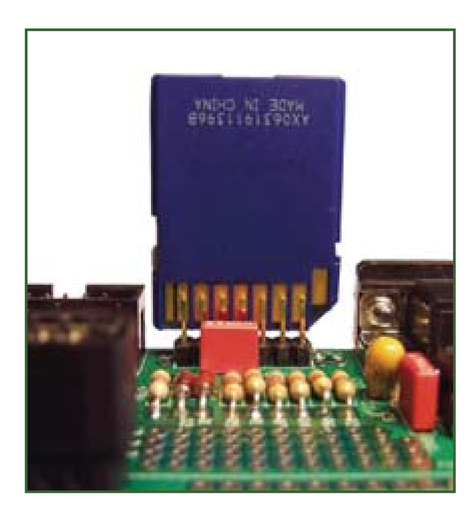


Figura 6. Una tarjeta SD conectada entre los pines del conector.



Figura 7. Un lector de tarjeta barato proporciona, tras su extracción, una ranura para la tarjeta SD.

Sobre los autores

Ulrich Radig y Holger Buss son apasionados desarrolladores de software y hardware de código abierto, y en su página web se ocupan, además de más desarrollos del servidor web AVR, de muchos otros proyectos.



Figura 8. Así se sueldan los pines de la ranura de la tarjeta SD a los pines del conector.



Figura 9. Los pines sin función del conector se sueldan para mayor estabilidad a la chapa que cubre la ranura de la tarjeta SD.

muestra en la **Figura 8**, se suelda directamente a las pines de la placa. El hecho de soldar la tira de pines exteriores a la chapa (**Figura 9**) le da una estabilidad mecánica adicional.
En las tarjetas SD vienen los archivos disponibles en la página web del proyecto. Es importante sobre todo el archivo “server.cfg“. En él hay que fijar la dirección IP con un editor de texto, de manera que los primeros tres bits coincidan con la dirección del router. La página web real se encuentra en el archivo “index.asp“. Como no es un servidor de gran potencia, las viejas tarjetas SD con algunos cientos de MB de capacidad son más que suficientes.
Para hacer una primera comprobación, hay que observar la salida de la interfaz serie mediante el hiperterminal. Los parámetros son: 9.600 baudios, 8 bits de datos, sin paridad, 1 bit de parada y sin control de flujo. Cuando se alcance una tensión de alimentación de entre 9 y 12 V y todo esté en orden, debería aparecer un resultado como en la captura de pantalla de la **Figura 10**. Además, se debería poder ver entre las direcciones IP seleccionadas la página web de la tarjeta SD.

Información adicional

El servidor funciona también sin tarjeta SD. En ese caso, hay que instalar de todos modos una resistencia de polarización adicional de 100 kΩ del pin 7 de la interfaz de tarjeta SD hasta +5 V para que el servidor advierta que no existe ninguna tarjeta SD instalada. Además, la resistencia no molesta en el funcionamiento con tarjeta SD. En el firmware 1.4 se determina sin tarjeta la dirección IP 192.168.1.20. Aunque seguro que el servidor no es el más rápido, hay que decir que responde a la ejecución del comando ping correctamente en 1 ms.
Más adelante [5] puede ver en vivo el prototipo del laboratorio ELEKTOR (**Figura 11**), pero, por favor, no todos a la vez porque fue concebido para poco menos de 1000 accesos por segundo ;-)
Para conectar periféricos (sensores de temperatura o cámara web) consulta la página web de los autores del proyecto, donde encontrarás información más precisa.

(060257e)

Enlaces:

[1] Proyectos de Ulrich Radig:
www.ulrichradig.de

[2] Proyectos de Holger Buss:
www.mikrocontroller.com

[3] Página web de proyecto del servidor web:
www.mikrocontroller.com/de/avr_webserver.php

[4] Página web de proyecto de ELEKTOR:
www.elektor.de/webserver

[5] Dirección del servidor web:
www.elektor.de/testserver/

[6] Foro sobre el servidor web:
www.ulrichradig.de/site/forum/viewforum.php?f=5

[7] Modificación para el chip 3C5x9:
www.mikrocontroller.com/de/ISA_3Com.php

[8] Ficha técnica del ATmega644:
www.atmel.com/dyn/resources/prod_documents/doc2593.pdf



```
------ AVR-Server V1.40 -------
MMC/SD: Ok. reading 'server.cfg'...Ok
..........
NIC  : Init RTL8019: Ok
MAC  : 01 0A 12 24 00 FF
IP   : 192.168.1.102
----
ROUT: 192.168.1.0
NTP  : 132.187.3.3
FTP  : 192.168.1.81
SMTP: 192.168.1.81
----
NTP Enabled=NO
----
Mail-Trigger Enabled=NO
SMTP AUTH LOGIN=NO
CAM Enabled=NO
CAM Store SD-Card Enabled=NO
FTP Push Enabled=NO
CAM Store (SD or FTP) Time=   5
External Trigger Enabled=NO
```

Figura 10. Cuando veas en el hiperterminal un resultado parecido, lo habrás hecho todo correctamente.



Figura 11. Así se anuncia el prototipo del servidor web del laboratorio de ELEKTOR.

Interfaces USB-UART

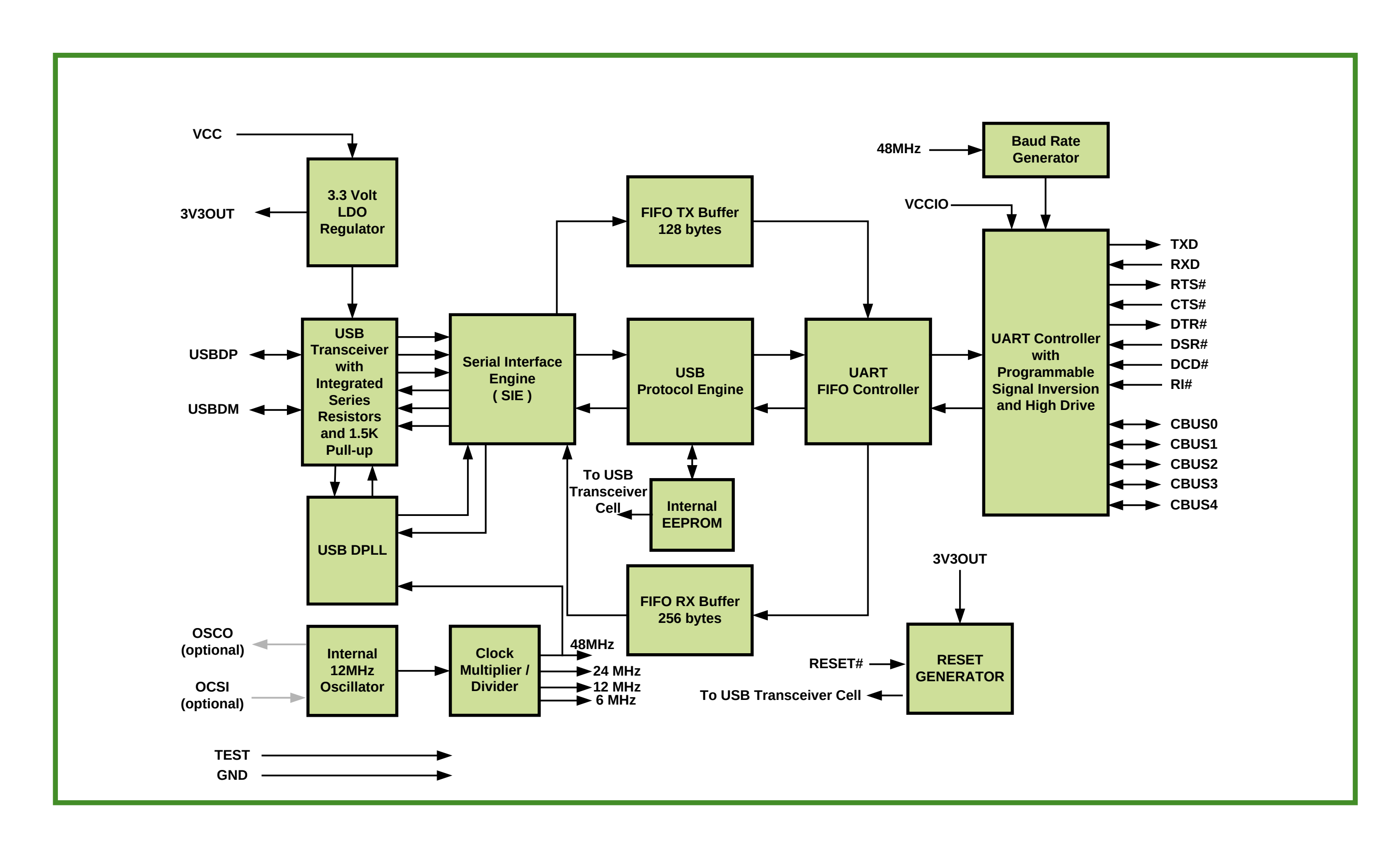
Todo lo que debe saber para sobrevivir

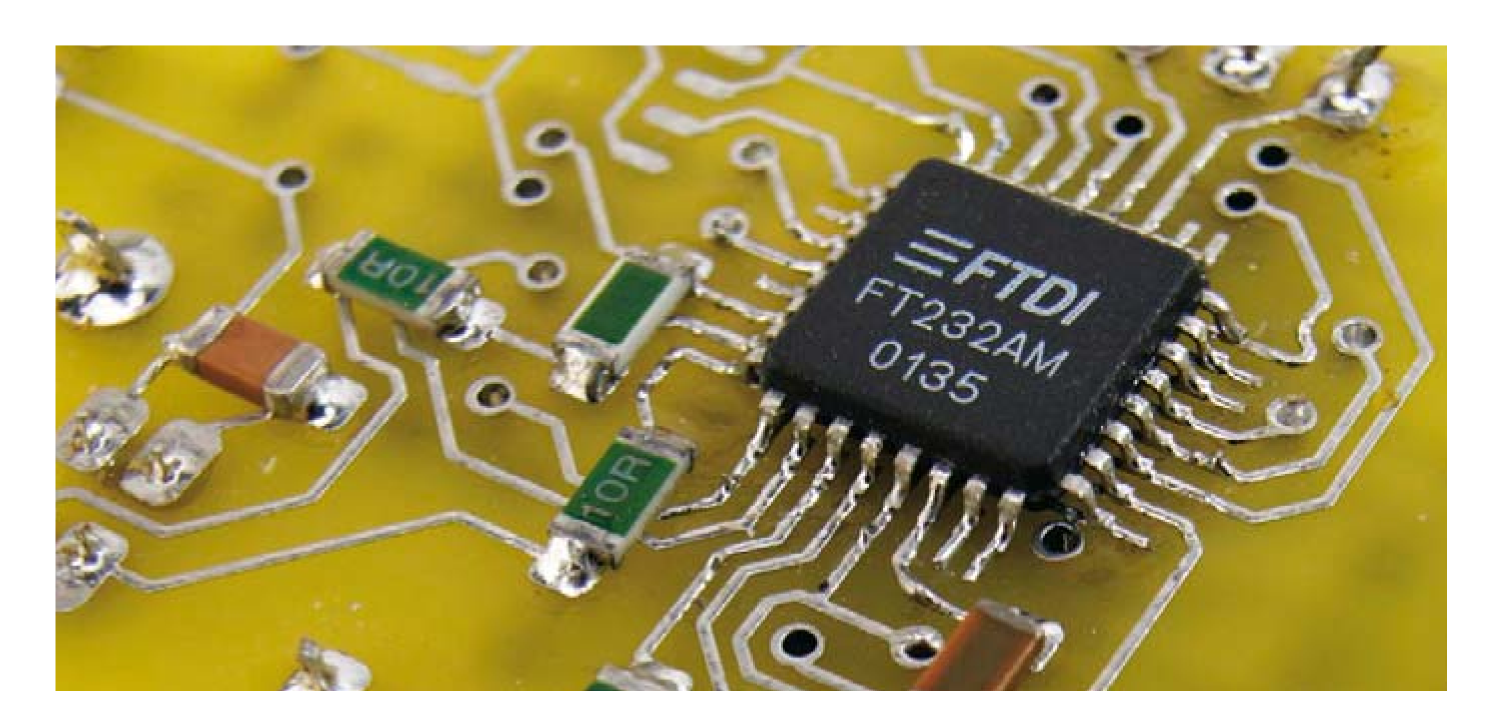
Antoine Authier

**Los primeros chips de interfaz entre el bus USB y los bus de serie asíncronos (UART) aparecieron a principios de la década de 2000.
Y ya estamos en la tercera, cuarta, o incluso quinta generación de circuitos: más rápidos, más fiables y más estables (concretamente sus drivers), y con funciones mejoradas.**

Estos circuitos integrados permiten conectar a un bus USB maestro cualquier aparato que requiera una conexión de serie asíncrona *full duplex*, compatible o no con la norma RS-232.

En general, los drivers presentan dos modos de uso. El primer modo permite simular un puerto COM tradicional en la mayoría de sistemas operativos (Windows, Linux). Así, flexibiliza la transición tecnológica facilitando el uso transparente en el bus USB de materiales antiguos concebidos para los puertos de serie.
El segundo afecta a las transferencias de datos a alta velocidad (es decir, transferencia directa de bloque, *bulk mode*), especialmente gracias al uso de un buffer de mayor tamaño, a condición de que pueda abandonarse la señalización RS-232 clásica. Algunos circuitos permiten superar tranquilamente la línea de 115.200 bps. Los tres principales actores del mercado son el escocés *Future Technology Device International Ltd* (por si jamás se habían preguntado qué se escondía detrás del acrónimo FTDI), el americano *Silicon Laboratories Inc*. (*Silabs* para los amigos), y el taiwanés *Prolific Technology Inc*.





FTDI

Los dos productos más recientes de FTDI son el FT232R y el FT2232D.
El FT232R presenta una interfaz USB de serie UART (*Universal Asynchronous Receiver/Transmitter)*, y funciona también en modo *bit bang* síncrono; este modo, específico de los chips FTDI, les permite controlar 8 entradas/salidas (E/S) de alta velocidad. La EEPROM de configuración, las resistencias de conexión al bus USB así como el circuito de reloj se han integrado en el chip para simplificar el uso y reducir la congestión.
La señal de reloj interna, configurable (6, 12, 24 o 48 MHz), es disponible en uno de los terminales del circuito. También puede usarse para establecer la frecuencia de un microcontrolador o la lógica externa.

Durante la producción, se atribuye un número de identificación único (*FTDI-Chip-ID*) a cada circuito. Esta innovación técnica, que constituye el orgullo de FTDI, permite por ejemplo proteger una aplicación contra la copia.
El FT2232D presenta dos interfaces USB UART/FIFO (*First In-First Out*) configurables independientemente. Concretamente, soporta los siguientes modos: interfaz UART, FIFO, *bit bang*, interfaz de serie síncrona compatible JTAG, I^2C y bus SPI. Estas tres últimas interfaces forman un circuito de tomas para conexiones directas en una amplia gama de chips. La compatibilidad JTAG facilita la depuración y la programación sin fallos.

FTDI suministra los drivers de estos circuitos sin gastos ni royalties, tanto para los sistemas operativos de Microsoft como para Linux (núcleo 2.4.x únicamente, que podría parecer antiguo en el momento de escribir estas líneas).
Bill Ryder es el iniciador del COM virtual para Linux, http://ftdi-usb-sio.sourceforge.net/ y coordina el desarrollo a largo plazo. Su driver está integrado en el núcleo desde la versión 2.4.20. Es responsable de la creación del periférico `/dev/ttyUSBx` en el empalme de un circuito FTDI en *Linux*.
Otros colaboradores han escrito otras implementaciones y complementos. *Intra2net* por ejemplo, proporciona una biblioteca específica para el acceso en modo *bit bang*, http://www.intra2net.com/de/produkte/opensource/ftdi/. Sobre la base de la *libusb* estándar, esta biblioteca funciona muy bien, por lo menos por lo que hemos podido comprobar.
Los drivers para *Windows* están disponibles en el sitio y bien documentados.
Parece que FTDI va a la cabeza de los circuitos más usados en los proyectos de electrónica que se hace uno mismo.
Se ha publicado mucho savoir-faire al respecto, así que es beneficioso para todos.
Aunque es cierto que el suministro era complicado hace unos años, ahora la red de distribución está bien desarrollada y los chips FTDI están fácilmente disponibles y a precios razonables.
Pronto les presentaremos su última innovación, el *Vinculum*, un microcontrolador con puerto USB maestro.

Testimonio de un usuario

"El control del flujo material no funcionaba correctamente. Por suerte para nosotros, el soporte técnico de EE.UU. corrigió muy rápidamente este problema.

Por el contrario, con el driver para Linux no se suministra ningún código fuente. Así que para la adaptación tuvimos que hacer un poco de bricolaje y de descompilación.

En cuanto a la sincronización del puerto de serie, el USB fuerza la transmisión de las tramas por paquete, lo que provoca un ligero retraso, de algunos milisegundos si no recuerdo mal, tanto en la emisión como en la recepción.

Una cosa es cierta, esto no le impide alcanzar velocidades de transmisión mucho más elevadas que las de un puerto de serie clásico".

Frédéric, R&D Inventel, París

Tabla comparativa			
	FTDI		Silicon Laboratories
Características– especificidades	FT232R	FT2232D	CP2101
Norma USB	2.0 Full Speed	2.0 Full Speed	2.0 Full Speed
Tamaño de la EEPROM de los descriptores USB (en octetos)	1024	externa	512
Tamaño del buffer de emisión (en octetos)	128	128	512
Tamaño del buffer de recepción (en octetos)	256	384	512
Número de bits de datos soportados	7 y 8	7 y 8	8
Número de bit de stop soportados	1, 2	1, 2	1
Paridad soportada	Par, impar, ninguna marca, espacio	Par, impar, ninguna marca, espacio	Par, impar, ninguna
Conformidad de conexión (Handshaking)	Material o software X-On X-Off	Material o software X-On X-Off	Material y software
Velocidad de transferencia (Baud rate)	RS232: 300 bps a 1 Mbps RS485: 300 bps a 3 Mbps	RS232: 300 bps a 1 Mbps RS485: 300 bps a 3 Mbps	300 bps 921,6 kbps
Interrupción de línea (Suspend & Resume)	Sí	Sí	No
Velocidades de transferencia parametrizadas (no estándar)	Sí	Sí	No
Período de espera en recepción parametrizable	Sí	Sí	No
Puertos de entrada/salida genéricas	5 configurables	configurable	No
Tensión de los puertos de entrada/salida configurable por terminal Vio	1,8, 2,8, 3,3, 5 V	3,3, 5 V	No
Puesta en modo de espera dinámica por puerto USB		Sí	No
Soporte de protocolo RS-232	Sí	Sí	Sí
Soporte de protocolo RS-422	Sí	Sí	No
Soporte de protocolo RS-485	Sí	Sí	No
Otro bus de serie	–	JTAG, I2C, SPI	–
Modo de transferencia de bloque	Sí, bit bang	Sí, bit bang	Sí
Reloj interno	6, 12, 24, 48 MHz	No	–
Adaptación de impedancia del bus USB integrado	Sí	No	Sí
Tensión de alimentación	3,3, 5,25 V	4,35, 5,25 V	3,0, 3,6 V
Alimentación por el bus USB	Sí	Sí	Sí
Número de serie	FTDIChip-ID	No	No
Caja	QFN-32 SSOP-28	LQFN-48	QFN-28
Pilotes (drivers) logiciels	Port COM Virtuel (W/L) D2XX (W/L)	Port COM Virtuel (W/L) D2XX (W/L)	Port COM Virtuel (W/L) USBXpress (W)

Silicon Laboratorios

En la gama de las interfaces USB-serie, *Silicon Laboratories* propone tres circuitos: los CP2101, CP2102 y CP2103.

El CP2101 constituye la base y la referencia. Presenta todas las funciones de base de la interfaz USB-UART en *Silabs*. Su caja de 5 mm incluye un controlador USB 2.0 *full speed*, un reloj, el circuito de adaptación de impedancia para una conexión sin resistencias en el bus USB, una EEPROM de 512 octetos y un controlador UART compatible RS-232. Así, no es necesario ningún componente externo dedicado al bus USB.

Tanto el CP2102 y el CP2103 son mejoras consecutivas del anterior, con un mayor número de funciones. La EEPROM interna cuyo tamaño ha pasado a 1024 bits puede protegerse con escritura. Al CP2103 se le han incorporado 4 puertos de E/S independientes, cuya tensión de salida es regulable (por medio de una alimentación externa). Soporta también el protocolo RS-485 y la puesta en espera dinámica a través del bus USB.

Silicon Laboratories propone también dos tipos de drivers. La emulación de puerto COM está disponible tanto para *Windows* como para *Linux*. En esta configuración, la interfaz UART soporta todas las funciones del protocolo RS-232 así como la conformidad de conexión (*handshaking*). Para el CP2103 también se utiliza el protocolo RS-485 y manda cuatro GPIO (*General Purpose Input/Output* = E/S de uso genérico). Un driver llamado *USBXpress*, disponible únicamente para *Windows*, accede directamente al chip.

Es interesante precisar que, por desgracia, el driver para *Linux* no se suministra con su código fuente.
No obstante, Craig Shelley ha reescrito este driver, que está disponible con las

		Prolific		Moschip	
CP2102	CP2103	PL2303	PL2313	MSC7820	MSC7840
2.0 Full Speed	2.0 Full Speed	2.0 Full Speed	1.1	2.0 Full Speed	2.0 Full Speed
1024	1024	OTP + externa	OTP + externa	Externa I2C	Externa I2C
640	640	256 / 128	–	512	512
576	576	256 / 328	–	512	512
5, 6, 7 y 8	5, 6, 7 y 8	5, 6, 7 y 8	–	5, 6, 7 y 8	5, 6, 7 y 8
1, 1,5 y 2	1, 1,5 y 2	1, 1,5 y 2	–	–	–
Par, impar, ninguna marca, espacio	Par, impar, ninguna marca, espacio	Par, impar, ninguna marca, espacio	–	–	–
Material y software	Material y software	Material o software X-On X-Off	Material y software	Material y software	Material y software
300 bps 1 Mbps	300 bps 1 Mbps	75 bps a 6 Mbps	Hasta 1 Mbps	50 bps a 6 Mbps	50 bps a 6 Mbps
Sí	Sí	Sí	–	Sí	Sí
Sí	Sí	–	–	Sí	Sí
Sí	Sí	–	–	–	–
No	4	2	12	1	1
No	1,8 V a Vdd	–	–	–	–
No	Sí	Sí	–	Sí	Sí
Sí	Sí	Sí	Sí	Sí	Sí
No	No	No	–	Sí	Sí
No	Sí	No	–	Sí	Sí
–	–	–	Sí (japonés)	–	–
Sí	Sí	Sí	–	–	–
–	Sí	No	No	No	No
Sí	Sí	Sí	Sí	–	–
3,0, 3,6 V	3,0, 3,6 V	3,6, 5,5 V	3,3, 5,5 V	4.5, 5.5V	4.5, 5.5V
Sí	Sí	–	–	–	–
No	No	Sí	–	–	–
QFN-28	QFN-28	QFN-32 SSOP-28	LQFP-48 SSOP-28	LQFP-48	LQFP-64
Port COM Virtuel (W/L) USBXpress (W)	Port COM Virtuel (W/L) USBXpress (W)	Port COM Virtuel (W/L) Accès direct (à tester)	–	Windows/Linux/Mac	Windows/Linux/Mac

fuentes del núcleo actual. Todavía está en estado de desarrollo avanzado.
Por el contrario, el sitio de *Silabs* propone una base de conocimiento rica y detallada. Los circuitos de aplicación se proporcionan con el código fuente de su microprograma (*firmware*), y son de buena calidad.

Prolific

Prolific nos propone dos circuitos recientes: el *PL2303X-Edition* y el PL2313 que proporcionan respectivamente una o dos interfaces USB de serie.
El PL2303X respeta las especificaciones del bus USB 2.0 *full-speed*. El circuito presenta un reloj interno a 12 MHz y dos puertos de entrada/salida genéricos. El UART utiliza el protocolo RS-232.
Una memoria OTP permite configurar los parámetros del enlace de serie y los datos del constructor, *VendorID* y *ProductID*. También se puede introducir un número de serie en esta memoria.

Esta información se puede modificar en la EEPROM externa cuando ésta se utiliza.

El PL2313 parece que está basado en un corazón más antiguo, responde a la norma USB 1.1. Presenta dos interfaces USB-UART que respetan el protocolo RS-232. Contiene un reloj interno, las resistencias de bus, una memoria OTP para la personalización y doce puertos de entrada/salida.
Además, es compatible con varias interfaces del teléfono móvil japonés *iMode*, CDMA-1...
Se recomienda adquirir los drivers para *Windows* y preinstalarlos para preparar el terreno. Sino, no funcionará la automatización de la instalación.

Las fichas de características de los componentes *Prolific*, los códigos fuente y las notas de aplicación están disponibles en la rúbrica [*support* > *do-*



wnload center] de su sitio. Proporcionan los drivers para *Windows*, COM virtual y acceso directo. Los ejemplos del código fuente describen el acceso directo para *Linux*. El driver de puerto COM para *Linux* ha sido desarrollado de forma independiente por Grez Kroah-Hartman, el código está integrado en el núcleo.

Los demás

El último fabricante que trataremos aquí es indio y presenta una gama llena de productos.

Las dos interfaces USB de serie recientes de *Moschip Semiconductor Technology Ltd* se llaman MCS7820 y MCS7840. Proporcionan respectivamente dos y cuatro interfaces que respetan la norma USB 2.0 y los protocolos RS-232, RS-422 y RS-485. Estos componentes requieren una EEPROM compatible I²C y pueden alimentarse directamente por el bus USB. Los drivers están disponibles tanto para *Windows* como para *Linux*. Todavía no hemos encontrado este tipo de productos en Europa.

K-micro, un fabricante americano, también ofrece este tipo de producto, pero no consigue convencernos, así que nos limitamos a mencionarlo, y dejamos al lector la libertad de seguir leyendo en su sitio.

Al término de esta presentación, tenemos la impresión de que cada fabricante propone una gama completa de productos, más o menos idénticos. Los precios, el soporte ofrecido y la información disponible de los fabricantes, así como la tabla recapitulativa que hemos elaborado, deberían facilitar la elección.

Observaciones y recomendaciones

- (probado únicamente para FTDI) Atención, la instalación conjunta del driver COM virtual y del acceso directo D2XX no funciona, le pedirá que proceda a la desinstalación del driver actual.

- Si su circuito propone dos interfaces, deberá instalar dos veces el mismo driver. En efecto, cada interfaz tiene su propio grupo de End-Point USB gestionado separadamente por el maestro.

(070478-I)

Enlaces Internet:

www.ftdichip.com/
www.silabs.com/
www.prolific.com.tw/
www.k-micro.us/
www.compuapps.com/
www.moschip.com/
www.usb.org/

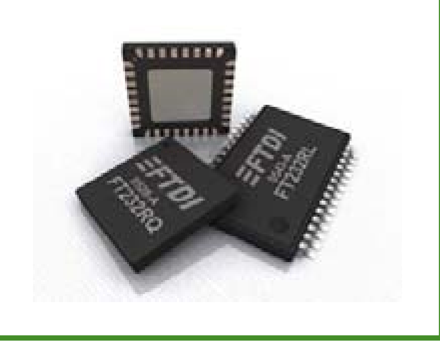
Personalice sus productos

✔ Modificar la EEPROM de los circuitos para cambiar el VendorID y/o el ProductID de su aparato puede resultar peligroso. Los drivers de referencia ya no reconocerán el aparato.

✔ Si desea personalizar sus montajes y poseer su propio VendorID oficial, sepa que cuesta 2.000$ ante la autoridad de regulación: www.usb.org/

✔ No obstante, puede procurarse un lote de diez VendorID/ProductID únicos y oficiales en www.mecanique.co.uk/ por la módica suma de 29,95 libras esterlinas.

✔ Por otro lado, FTDI ofrece a las empresas la posibilidad de tener un grupo de ocho ProductID a condición de que se utilicen junto con el VendorID de FTDI (a saber 0x0403).



Explorador CAN

Interfaz PC-CAN cómodo

Del ingeniero (ET) Fredi Krüger

Un bus CAN no es adecuado sólo para los coches y la automatización, también se puede incorporar al control doméstico. Por desgracia, no es del todo fácil montar un sistema bus así. Aquí llega nuestra placa CAN, que conecta un PC al bus. Con un programa cómodo y gratuito se puede supervisar un bus CAN y verificar la función de otros nodos.

Las redes que se basan en el bus CAN no sólo se emplean en coches y la industria (automatización, control), sino también, y cada vez más, en el ámbito privado. En internet hay una buena muestra de ejemplos interesantes [1]. Las ventajas fundamentales de estos buses de dos líneas son elevada inmunidad frente a las interferencias y una amplia gama de componentes adecuados (en [2a] y [2b] hay una introducción, en [2c] un resumen de las especificaciones técnicas). Los componentes de nodos disponibles abarcan desde los llamados controladores independientes (Stand Alone) hasta microcontroladores con un controlador CAN integrado. La dificultad de la realización de sistemas de bus CAN radica en realizar una comunicación entre los componentes independientes, porque se pueden tratar diversos parámetros. Todo se complica más cuando hay que construir un bus desde cero y programar posibles emisores CAN, así como los receptores correspondientes. Aquí entra nuestra interfaz CAN-PC, que conecta un ordenador con Windows al bus. Con ayuda del programa gratuito correspondiente, no sólo se puede supervisar con comodidad un bus ya existente, sino que también se pueden enviar y recibir comunicaciones CAN y así verificar la función de otros nodos CAN. Es evidente que así se puede obtener mucha información.

Fácil, polivalente, económico

Como controlador CAN se utiliza el componente MCP2515 de Microchip, que tiene un precio muy asequible. El programa de PC correspondiente, que también es de Microchip, incluso es gratuito [3]. Con el programa no sólo se pueden introducir los registros del controlador CAN para adaptarlos a la aplicación correspondiente, también se pueden configurar notificaciones para a enviar y un filtro de notificaciones para aquellas que haya que reenviar.

El control de la placa se realiza mediante el interfaz paralelo del PC (puerto de impresora). Aunque hoy en día apenas se utiliza ya para manejar impresoras, todavía muchos PCs disponen de el. Con tarjetas de memoria muy económicas también se puede actualizar un interfaz así (para posibles problemas con Windows Vista véase nuestros Rumores de laboratorio de este número). El interfaz paralelo tiene que estar configurado para el tráfico bidireccional de datos.

Así, el circuito (véase la **Figura 1**) también está adaptado para programas que puedan utilizarse más tarde y se corresponde en gran medida a la descripción que hay en la página web de Microchip [3]. Para proteger el puerto de impresora se conectan todas las líneas de señal utilizadas a dos 74HCT245 (IC1, IC2), antes de conectarlos con el controlador CAN MCP2515 (IC3). La transmisión se produce en serie por el interfaz SPI y mediante el programa de PC las señales correspondientes están disponibles en el puerto de impresora. Las otras señales que se emplean sirven para la transmisión o consulta de conexiones adicionales del controlador CAN. Como controlador de bus para la conexión al bus CAN en sí se utiliza el componente MCP2551 (IC4). Sin embargo, también se pueden introducir componentes parecidos, como un PCA82C521.

También en la placa de circuito impreso, disponible en Elektor (ref. 060201-91), ya están montados IC3 y IC4 en sus variantes SMD (de IC4 hay todavía otra forma

Figura 1. Para proteger el puerto de impresora se conectan todas las señales aa dos 74HCT245 (IC1, IC2), antes de conectarlo con el controlador CAN (IC3). Como controlador del bus para la conexión al bus CAN se utiliza IC4.

disponible). Con el puente JP1 se puede poner una resistencia de terminación de 120 ohms para evitar reflexiones e interrupciones de la transmisión. Para una conexión fácil del bus CAN está previsto un conector de 9 pines (K2) en la placa. La posición correcta para la conexión se puede ver en el diagrama de circuito. También está prevista la posibilidad de alimentar otros componentes mediante esta conexión. Con el puente JP2 se puede seleccionar entre los 5 V del regulador de tensión situado en la placa o una conexión directa a la fuente de alimentación conectada a la placa, y en el último caso un puente rectificador sirve como protección contra la inversión de voltaje o para rectificar una tensión alterna.

Montaje y puesta en marcha

El montaje empieza por las resistencias y los condensadores, el resto de componentes se montarán según su tamaño. Hay que tener en cuenta la correcta polaridad de los dos condensadores electrolíticos (C7, C10), el LED (D1) y el rectificador (B1), así como la posición del IC. Después de montar todos los componentes en la placa y realizar una última comprobación visual, se puede conectar la tensión de alimentación en K3, que debería estar entre los 8 V y 12 V aproximadamente. Ahora debería iluminarse el diodo LED D1. Si la placa está conectada con el PC mediante el interfaz de impresora y se accede al programa descrito a continuación, debería reconocerla automáticamente como placa de evaluación.

Programa

El programa se puede descargar de forma gratuita de la página web de [3]. Se basa en el programa CANking de la empresa Kvaser [4], pero con el controlador adecuado para el módulo MCP2515. El programa se puede utilizar con todos los sistemas operativos de Microsoft desde Windows 95. Se necesita el controlador “dlportio.sys“, que se encuentra en:

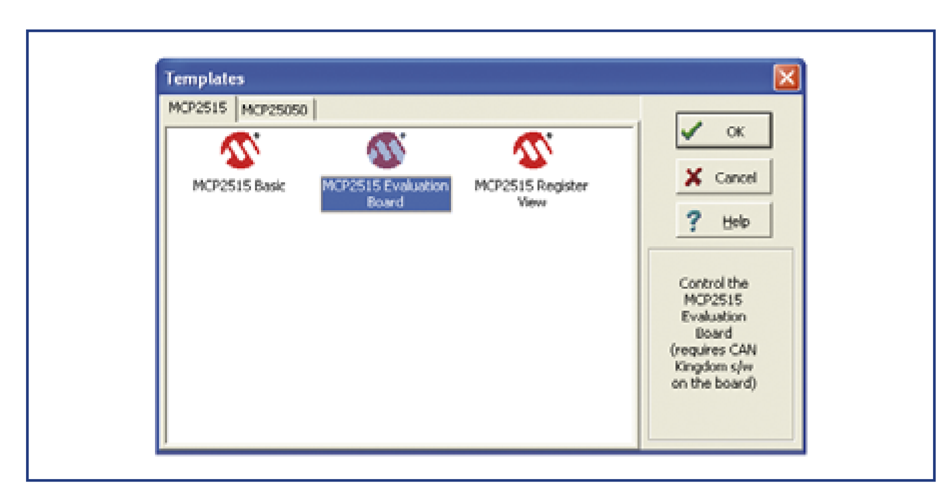
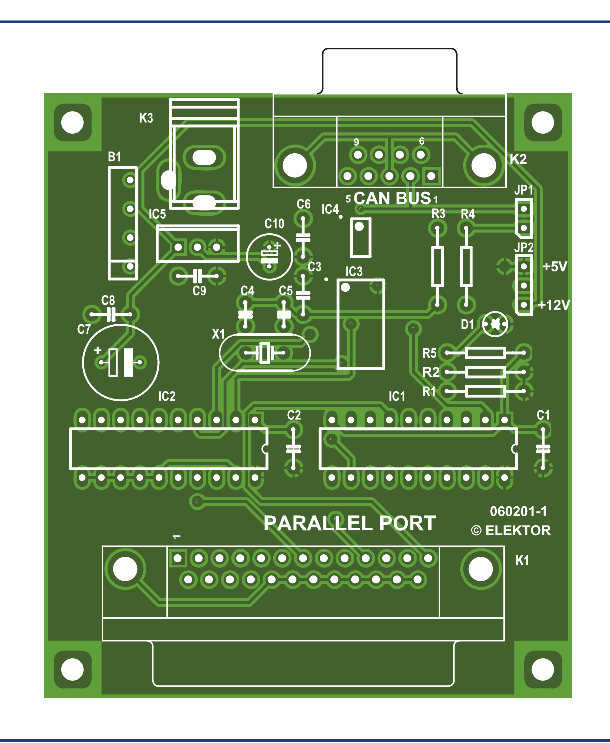
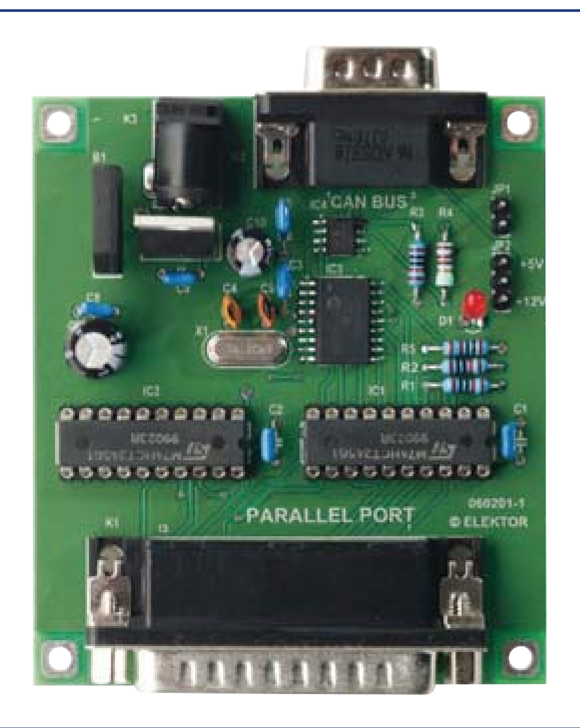


Figura 2. Después de la instalación y algunos clics aparece esta ventana de selección.





Lista de componentes

Resistencias (SMD 1206):
R1,R2,R3 = 10k Ω
R4 = 120 Ω
R5 = 1k Ω

Condensadores:
C1,C2,C3,C6,C8,C9 = 100n (SMD 1206)
C4,C5 = 22p (SMD 0805)
C7 = 470µ/25V (radial)
C10 = 47µ/16V (radial)

Semiconductores:
IC1,IC2 = 74HCT245
IC3 = MCP2515
IC4 = MCP2551SN (SMD SO-08), o PCA82C250/PCA82C251 (IC3 e IC4 ya están montados en la placa de Elektor)
IC5 = 7805 (TO220)
B1 = B80C1500 (rectificador)
D1 = LED 3 mm/2 mA

Varios:
JP1 = tira de 2 pines con puente
JP2 = tira de 3 con puente
X1 = cuarzo 16 MHz (HC49/S)
K1 = Conector SUB D de 25 pines, para montaje en placa
K2 = Conector SUB D de 9 pines, para montaje en placa
K3 = Conector CC hembra (HEBW25)
Placa (disponible con el número 060201-91 con IC3 y IC4, véase www.elektor.es)

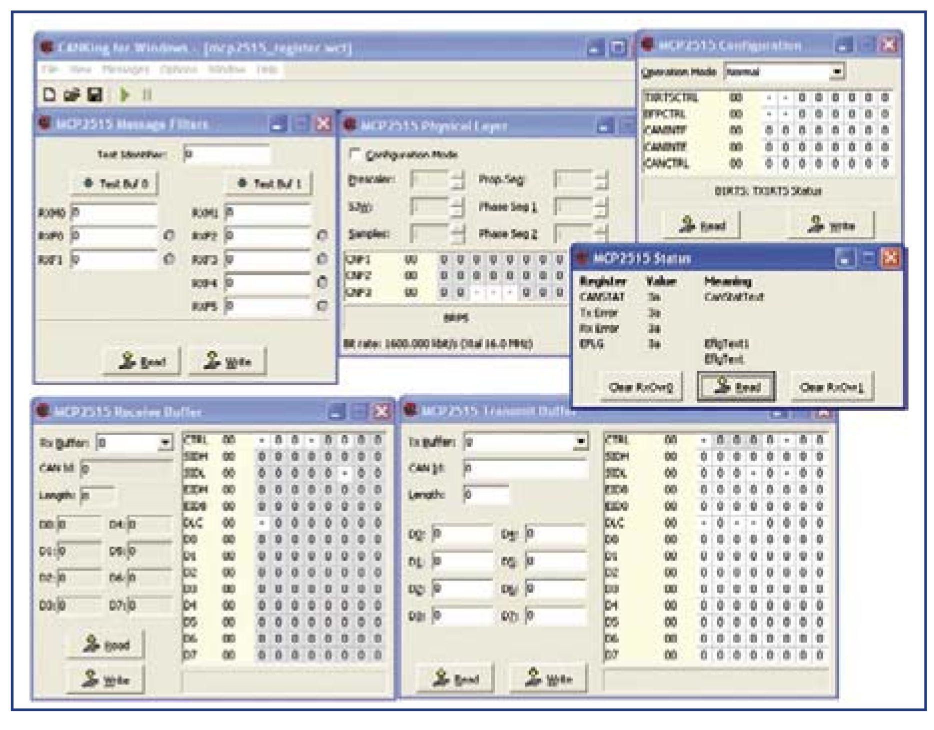


Figura 3. En "MCP2515 Register View" se tiene acceso a todos los registros del módulo MCP2515.

"C:\Windows\system32\drivers" o algo parecido. Quien no lo encuentre puede descargarlo de forma gratuita en [5]. Después de descargarlo, primero hay que instalar el programa en el ordenador. Si se accede a él, después de algunos clics aparece una ventana de selección como la de la **Figura 2**. De todos modos, en algunos PCs el controlador dlportio.sys no se iniciará de forma automática, entonces se recibe un mensaje de error y hay que iniciar el controlador manualmente. También es un requisito imprescindible para una buena conexión que esté introducida en el programa la dirección correcta del interfaz de impresora utilizado. Encontrará indicaciones más precisas sobre estos dos puntos en el documento "Fehlerquellen_Test.pdf", que se puede descargar de la página web de Elektor [6].
Si se selecciona "MCP2515 Register View" se tiene acceso a todos los registros del módulo MCP2515. Cada función tiene su ventana correspondiente (**Imagen 3**).
Si se selecciona "MCP2515 Evaluation Board" o "MCP2515 Basic", el programa

actúa como un monitor del bus CAN que reproduce todas las actividades del bus (**Figura 4**). También se puede descargar en Microchip una extensa documentación [3] sobre el programa.

(060201jn)

Enlaces

[1a] www.mikrocontroller.net/articles/CAN_als_Hausbus
[1b] www.isysbus.org
[1c] www.canathome.de
[1d] www.iuse.org
[2a] www.me-systeme.de/canbus.html
[2b] www.hehlhans.de/bilder/autos/cdi270/can.htm
[2c] http://de.wikipedia.org/wiki/Controller_Area_Network
[3] http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeld=1406&dDocName=en531891
[4] www.kvaser.com
[5] www.driverlinx.com/download/dlportio.htm
[6] www.elektor.es

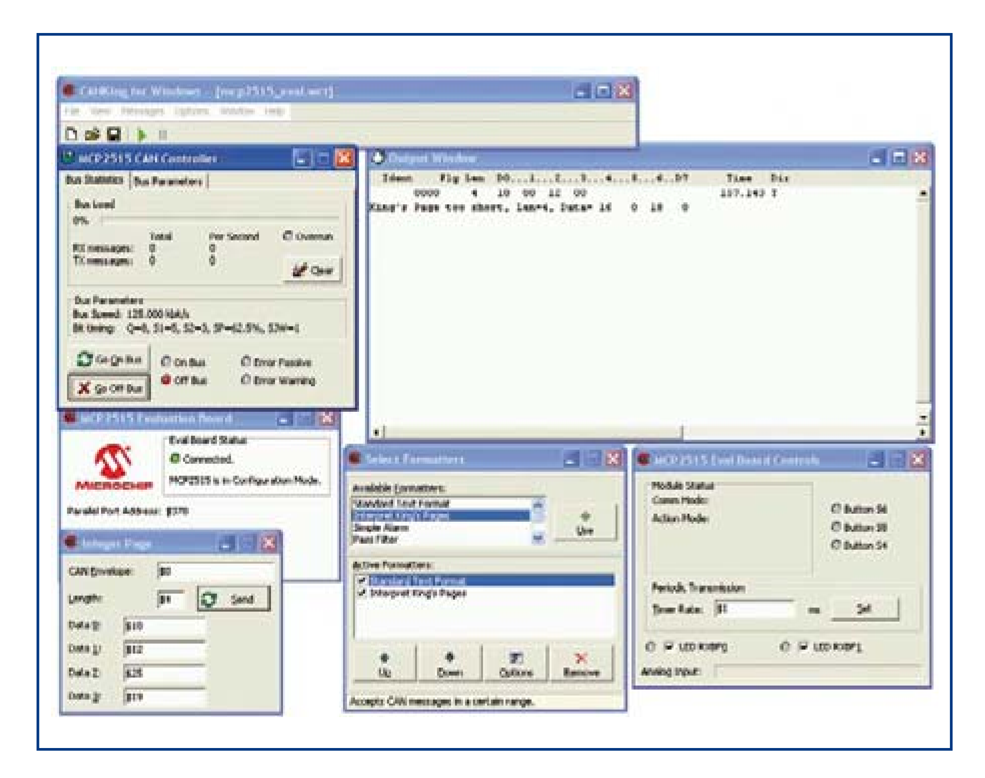
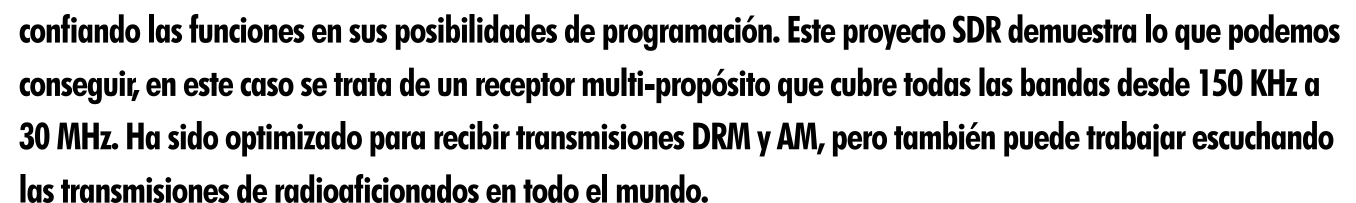


Figura 4. Si se escoge en la ventana de selección (véase la Imagen 2) "MCP2515 Evaluation Board" o "MCP2515 Basic", el programa actúa como monitor del bus CAN que reproduce todas las actividades del bus.

Radio definida por pr

Con interfaz USB

Burkhard Kainka



Los receptores radio SD (del inglés Software Defined, es decir, Definido por Programa), utilizan un mínimo de circuitería, confiando las funciones en sus posibilidades de programación. Este proyecto SDR demuestra lo que podemos conseguir, en este caso se trata de un receptor multi-propósito que cubre todas las bandas desde 150 KHz a 30 MHz. Ha sido optimizado para recibir transmisiones DRM y AM, pero también puede trabajar escuchando las transmisiones de radioaficionados en todo el mundo.

El objetivo del diseñador de este proyecto era crear un receptor con una alta linealidad y una buena precisión de fase. El desarrollo se ha enfocado a las características más importantes para un receptor DRM de primera clase y el resultado final es un receptor con unas características de rechazo a interferencias sobresalientes. La recepción de las estaciones DRM, utilizando el programa DREAM, proporcionó unos valores de relación señal/ruido (S/NR) en torno a los 30 dB. El principio de diseño del receptor garantiza una respuesta de la curva de filtrado extremadamente plana, lo cual no sólo se aprecia muy bien sobre la señal DRM, sino también en la calidad de audio de las transmisiones recibidas de AM, cuyo sonido es casi tan bueno como el de una estación de FM en VHF. Merece la pena destacar que algunos transmisores que no cumplen con los anchos de banda normales, al menos tan rígidamente como deberían hacerlo, llegan a caer hasta la onda media (9 kHz) y la onda corta (10 kHz). Estas estaciones producen una mejora de sonido que los oyentes que utilizan receptores normales apenas aprecian (ya que sus filtros de IF limitan el ancho de banda y, en dicho proceso, hacen lo propio con la respuesta en frecuencia), al contrario que en el caso de las estaciones SDR, donde no supone ningún problema seleccionar un ancho de banda mayor. En los receptores definidos por programa todavía es mejor, ya que las posibilidades de una sintonía fina, proporcionadas por un programa decodificador que corre sobre un ordenador, nos permite determinar el ancho de banda deseado como filtros "notch" para la configuración del control de nivel automático (ALC), junto con los modos usuales de recepción para AM por medio de DRM y BLU (Banda Lateral Única) en OC (Onda Corta).
Pueden añadirse refinamientos adicionales para aplicaciones SWL (escuchas de onda corta). Si, por ejemplo, deseamos aumentar la sensibilidad en las bandas superiores de radioaficionados, podemos conseguirlo fácilmente utilizando dos entradas de antena conmutables y proporcionando un circuito preselector optimizado o un preamplificador en una de ellas. La propia placa de circuito impreso del receptor proporciona un conjunto de entradas de RF bastante básico que, sin embargo, es perfectamente adecuado para una recepción de estaciones de radio. Una antena de hilo de la longitud adecuada aumentará la fuerza de las señales recibidas por encima del nivel de ruido atmosférico, para asegurar que no perdemos prácticamente información.

Requerimientos del circuito

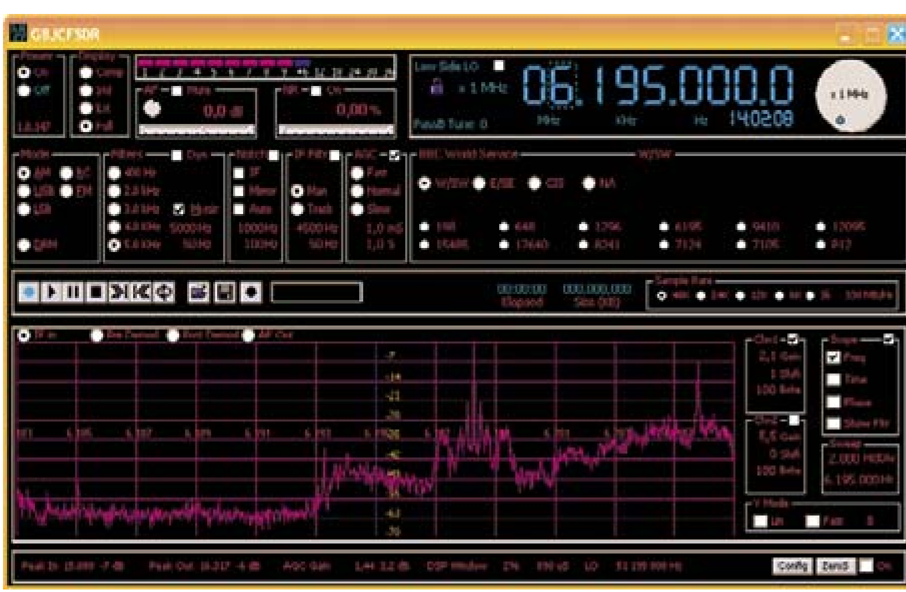
La mayoría de los programas SDR [1] necesitan la plataforma Windows XP para trabajar satisfactoriamente. Por lo tanto, la parte más importante del circuito es una tarjeta de sonido capaz de trabajar con señales SDR. Hemos desarrollado un pequeño circuito para verificar las tarjetas de sonido, que se describe en otro apartado de esta entrega bajo el título "Consejos para desarrolladores". ¡No tiene ningún sentido fabricar el receptor SDR sin hacer esta prueba!

Todo sobre USB

Una conexión USB controla al receptor, que está alimentado de la misma manera con una tensión de + 5 V (no necesita una tensión de red adicional). Para la interfaz USB, situada en el circuito receptor (ver **Figura 1**), hemos seleccionado el módulo FT 232R de nuestros amigos escoceses de la empresa FTDI. Este moderno conversor USB a Serie trabaja sin necesidad de un cristal de cuarzo, ya que está equipado con un oscilador RC interno que le proporciona la estabilidad adecuada. El módulo (IC4) se utiliza aquí en su modo "bit-bang" (un tipo de transmisión de datos serie que intercambia rápidamente un único bit de salida, "disparo de bit"), junto con las líneas de un puerto paralelo rápido. Pueden usarse ocho líneas de datos que se controlan en cualquiera de los modos que se desee. Dos de las líneas se utilizan como un bus I^2C y controlan la frecuencia del receptor. Tres hilos conectan la entrada del multiplexor a una de las ocho entradas de antena, con y sin filtrado. Hay dos entradas adicionales que sirven para controlar la amplificación de FI (Frecuencia

ograma





IC1 = 74AC74
IC2 = 74HC4066
IC5 = TL084CN
IC7 = 74HC4066

070039 - 11

Figura 1. Esquema eléctrico del circuito receptor que tan sólo comprende la parte del oscilador de sintonía y un mezclador.

Intermedia) en el receptor. De este modo, el receptor trabaja completamente bajo control remoto. ¡Adiós a todos los mandos y controles de las radios!
Debemos atender especialmente al desacoplo de la tensión de alimentación, porque el circuito integrado USB FT 232R trabaja internamente en el mismo rango de frecuencia que la señal que estamos recibiendo a través de la antena, y no es deseable que parte de esta señal de RF pueda pasar de una etapa a otra, a pesar de que el desacoplo dentro del propio circuito integrado es particularmente bueno y que la señal residual de RF en las líneas del puerto de control apenas es detectable. En consecuencia podemos controlar la entrada de RF del multiplexor en el circuito integrado HC 4051directamente desde las líneas del puerto de control, sin que haya restos de la señal de reloj del procesador en la zona de la señal deseada.
Utilizando su regulador de tensión interno de 3,3 V, el FT 232R proporciona la tensión de trabajo para el generador de reloj programable CY27EE16ZE, evitando la necesidad de un regulador de tensión adicional. El resto del circuito (**Figura 1**) opera exclusivamente con la tensión de + 5 V. Además, dentro del circuito se producen una serie de tensiones filtradas con su rizado atenuado que por un lado garantizan un buen desacoplo de RF, y por el otro aseguran una supresión de la interferencia de la frecuencia de audio. Este punto es particularmente crucial en la etapa de entrada de RF del receptor, desde donde se envía la señal, a través del mezclador, a la circuitería de FI. Por esta razón se ha colocado un condensador electrolítico de gran valor en este lugar (VCC_HF), de modo que se asegure una adecuada "paz y tranquilidad".

VFO programable

La radio definida por programa (SDR) hace una llamada a un oscilador de frecuencia que se ejecuta cuatro veces más rápido que la señal recibida, de manera que el filtrado de fase necesario puede dividirse entre cuatro. Si nuestro objetivo es recibir señales de hasta 30 MHz, el oscilador necesita llegar hasta una frecuencia de 120 MHz. Actualmente los osciladores DDS son muy populares en los proyectos de HF, pero a 120 MHz un DDS es bastante más caro, demanda un mayor consumo y es mucho menos controlable. Por este motivo hemos buscado una alternativa al oscilador DDS, optando por utilizar un reloj oscilador programable con un PLL interno. Muchos lectores de Elektor Electronics recordarán el CY27EE16ZE de nuestra entrega de abril de 2005. Este reloj oscilador, desarrollado especialmente para aplicaciones digitales, funciona con igual exactitud en circuitos de RF. La resolución de frecuencia no es tan buena como la de los osciladores DDS, pero la precisión de fase de la señal de salida alcanza resultados comparables. La reducción del consumo de potencia a una cantidad relativamente modesta es importante en este proyecto, ya que no podemos consumir demasiada corriente desde un puerto USB.
El circuito integrado se programa a través del bus I²C utilizando las líneas SCL y SDA. El VCO interno trabaja en el rango de frecuencias de 100 a 400 MHz, y está estabilizado por medio de un cristal de 10 MHz y un PLL. Así, su señal de salida pasa a través de los contadores hacia las salidas deseadas. En nuestro caso, hemos seleccionado la salida de reloj Clock5, donde podemos obtener una salida VFO comprendida entre 600 kHz y 120 MHz para su procesamiento posterior en el contador 74AC74.
El principio de funcionamiento del mezclador I-Q se ha descrito en la entrega de Elektor Electronics de febrero del 2007. Aquí hemos creado un mezclador de dos etapas a partir de un total de cuatro conmutadores analógicos situados en el interior del circuito integrado HC4066. Este circuito está controlado por dos señales del oscilador, desplazadas en fase, que se producen con un contador 74HC74. Si suponemos que el reloj oscilador programable produce una señal de 24 MHz, entonces el mezclador necesitaría una señal de control de 6 MHz. Así, el receptor podría en este caso trabajar en la región de unos ±24 kHz, a ambos lados de la frecuencia central de 6 MHz.
Lo importante aquí es que el desplazamiento de fase sea exactamente de 90° entre las dos señales del oscilador. Cualquier desviación afectará a la reducida supresión de las frecuencias imagen. En nuestro caso, cambiar el dispositivo de conmutación analógico por un 74HC4053 o un 74HC4052 no sería una buena elección, ya que el retardo producido por el paso de la señal en los decodificadores internos podría producir diferentes errores de fase que aparecerían en cualquier rango de frecuencia. La solución que hemos elegido utiliza conmutadores más básicos, como el HC4066, que mantiene las cuatro fases en sincronismo. Con el contador 74AC74 no deberíamos encontrar ningún error de fase en dicho circuito, de hecho el receptor presenta una supresión de la frecuencia imagen de unos 40 dB hasta los 15 MHz, aunque este valor disminuye más allá de los 20 MHz (lo cual es tolerable porque estas frecuencias no están altamente ocupadas).

Procesamiento de la señal

El receptor está equipado con varias entradas seleccionadas por el multiplexador de entrada 74HC4051 (IC6). La entrada de antena "ANT" es llevada, a través de filtros, a las tres primeras entradas. La selección del primer conmutador (ancho de banda) utiliza tan sólo una bobina de choque de entrada (L6), la cual elimina cualquier señal de entrada en la banda de audio. En la segunda posición del conmutador (Onda Media) hay un filtro paso/bajo con una frecuencia límite de 1,6 MHz , que utiliza la resistencia R12 para atenuar la resonancia excesiva. Este filtro suprime las interferencias en la recepción de Onda Media producidas por la mezcla de sobre-tonos con las estaciones en el rango de la Onda Corta. En la tercera posición el circuito hace uso de un sencillo filtro paso/alto RC que atenúa las señales fuertes de Onda Media.
También se puede seleccionar una señal de entrada adicional (PC1) si deseamos conectar circuitos de entrada externos sintonizados o preamplificadores. Por último, se han dejado libres tres entradas más para posibles desarrollos futuros (A4, A5 y A6). Los filtros de entrada que hay sobre la placa de circuito impreso son bastante buenos y adecuados para la mayoría de las aplicaciones. Por supuesto, podemos introducir un filtro paso/bajo delante de los filtros, siempre y cuando queramos realmente bloquear la mezcla de sobre-tonos en cualquier situación posible. También podemos optar por montar circuitos resonantes diferentes seleccionados mediante la electrónica de conmutación de entrada.
La entrada particular que está activa en un momento dado se conecta a la salida de un puerto COM (terminal 3). A ambos lados del conmutador se proporcionan condensadores de desacoplo, además de una línea de tensión de alimentación de 2,5 V para conmutar la conexión de la fuente del transistor BF 245 a través de una resistencia de 1 MΩ. Esto elimina cualquier distorsión proveniente de señales de entrada fuertes, que pueden aparecer cuando los diodos de protección en las entradas analógicas de los circuitos integrados limitan las señales.
La entrada A7 facilita una señal de calibración en la salida "Output 3" (Test-Clk) del oscilador de cristal programable. El oscilador produce una señal de onda cuadrada de 3,3 V pico a pico a una frecuencia de 5 MHz. En el divisor de tensión se produce una señal de tensión de unos 5 mV, a una frecuencia de 5 MHz, que se corresponde

LISTA DE MATERIALES

Resistencias
R1,R7,R19 = 100Ω
R2,R3 = 330Ω
R4 = 100Ω
R5,R8,R13,R17,R20 = 100kΩ
R6,R10,R14,R22 = 10kΩ
R9,R21 = 4kΩ7
R11,R23 = 27kΩ
R12,R15,R18 = 470Ω
R16 = 1MΩ
R24 = 1kΩ
R25 = 4Ω7

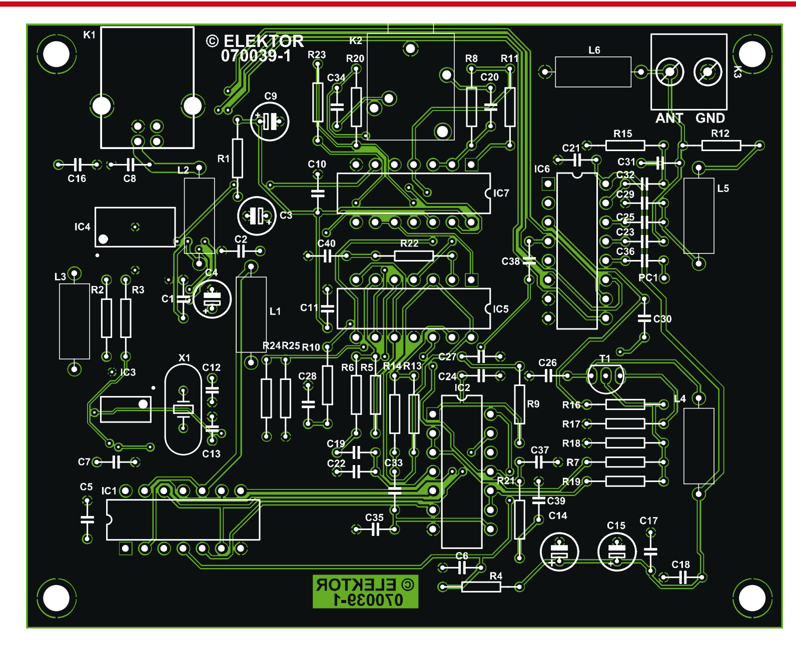
Condensadores
C1,C2,C5,C6,C7,C10,C11,C16,C17,C19,C20,C21,C25-C28,C30,C32,C33,C34,C36,C38,C39,C40 = 100nF
C3,C4,C9,C15 = 4μF7 condensador electrolítico de 16 V radial
C8,C18 = 10nF
C12,C13 = 10pF
C14 = 470μF condensador electrolítico de 16 V radial
C22,C24,C35,C37 = 2nF2
C29 = 220pF
C31 = 100pF

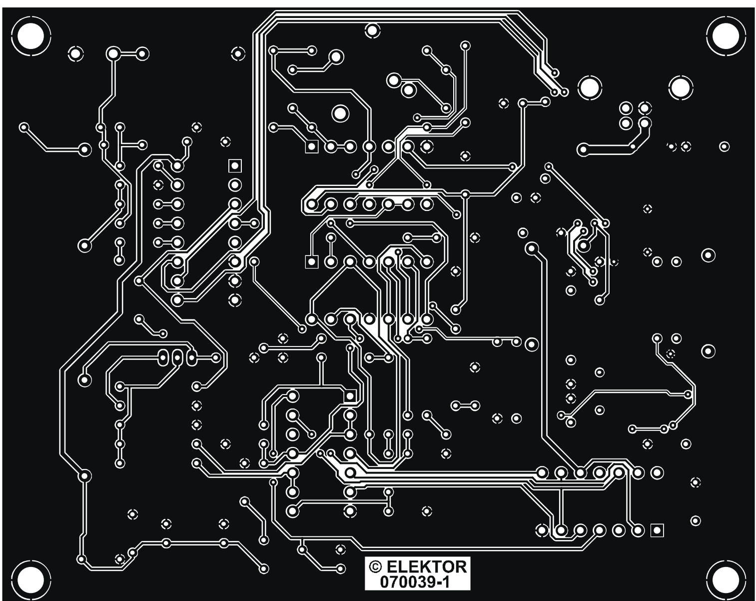
Semiconductores
IC1 = 74AC74
IC2,IC7 = 74HC4066
IC3 = CY27EE16 (Cypress)
IC4 = FT232R (FTDI)
IC5 = TL084CN con zócalo (ver texto)
IC6 = 74HC4051
T1 = BF245

Inductores
L1-L4 = 10μH
L5 = 47μH
L6 = 2.2mH

Varios
K1 = conector USB-B para montaje en placa de circuito impreso (PCI)
K2 = conector tipo “jack” estéreo, para montaje en placa de circuito impreso (PCI)
K3 = conector bloque terminal de 2 hilos para montaje en placa de circuito impreso (PCI), con separación entre terminales de 5 mm
PC1 = terminal de soldadura
PCI = Placa de Circuito Impreso totalmente montada y verificada con código de pedido nº 070039-91. Programas Ficheros con los programas del proyecto para su descarga gratuita con código nº **070039-11**.
Documentación complementaria, descarga gratuita de la placa de circuito impreso desnuda con código nº **070039-1**, desde www.thepcbshop.com

con una fuerza de señal de entrada de S9 de + 40 dB. Esto permite que el medidor de intensidad de campo creado en el programa pueda calibrarse sin ningún elemento adicional.
El transistor JFET BF 245 a la salida del





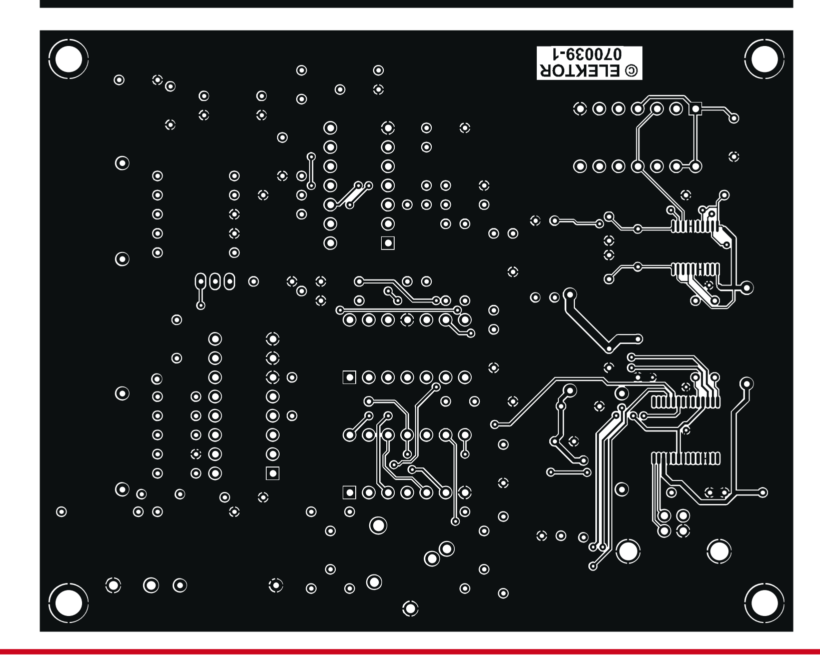


Figura 2. Placa de circuito impreso del receptor SDR.

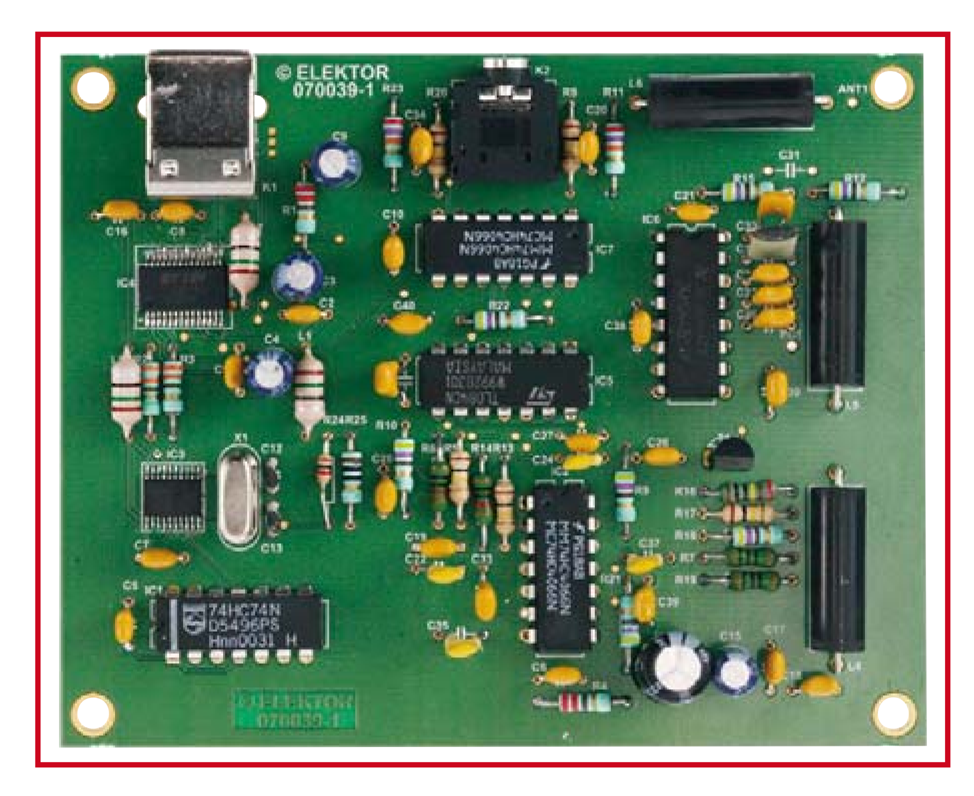


Figura 3. Esta placa ejemplo de laboratorio no es lo bastante equivalente para la inversión de producción suministrada a través de la tienda de Elektor.

multiplexador de entrada sirve como un transformador de impedancia. Esto proporciona una impedancia final relativamente alta de 100 kΩ para la señal de RF, lo que permite, por ejemplo, que pueda conectarse incluso un circuito resonante de alta Q a la entrada “In2”. Hemos intentado tener una tensión de 2,5 V aproximadamente a la salida de baja impedancia del seguidor fuente, la cual pasa a través del mezclador y del siguiente amplificador operacional y va directamente hacia la salida. Es importante que no aparezcan restos de señales en la frecuencia de audio en la conexión fuente, razón por la cual también se filtra minuciosamente la tensión de alimentación “crítica” Vcc_HF. El propio transistor FET proporciona el desacoplo adicional de la tensión de alimentación, pero no deseamos que ninguna señal escape de la puerta de dicho transistor, ya que dicha señal puede caer en la región de la FI, por debajo de los 24 kHz. Por este motivo se ha conectado una bobina de choque de RF directamente a la entrada de antena de forma que, por ejemplo, se elimine cualquier zumbido residual de la señal de red de 50 Hz.
Saliendo de la conexión fuente del transistor nos encontramos con dos resistencias de 100 Ω que van a los dos mezcladores para las señales I y Q. Dichas resistencias mejoran la simetría de los mezcladores, ya que la resistencia de activación de los mismos proporciona una cierta cantidad de fugas. Los propios mezcladores son circuitos integrados conmutadores analógicos del tipo HC4066, configurados como conmutadores de intercambio. La tensión de los mismos se selecciona también en torno a los 2,5 V, lo que permite controlarlos sin ningún tipo de sobrecarga hasta los 5 V pico a pico.
El amplificador de FI (Frecuencia Intermedia) está formado por dos ramas exactamente iguales que juntas producen una atenuación de hasta 40 dB en todo momento. Cuando estamos utilizando una tensión de alimentación de 5 V, la ganancia de ancho de banda (GBW) del amplificador operacional seleccionado es bastante importante, de manera que se pueda conseguir la amplificación (dividida en diez partes), sin errores de fase para señales que están en torno a los 20 kHz. En las muestras de las pruebas realizadas por el autor de este artículo se comprobó que un amplificador operacional TL 084 era adecuado para este trabajo. Si proporcionamos un zócalo para IC5 podríamos intentar probar con otros amplificadores operacionales más rápidos.
La etapa de entrada trabaja como un amplificador diferencial. Los valores de las resistencias que buscamos no son los más aptos para una mejor supresión en el modo común, sino los que hacen la resistencia de entrada lo más igual posible entre las entradas inversoras y no inversoras de los amplificadores operacionales. Las pruebas demuestran que una buena precisión de fase (y en consecuencia una alta supresión de la frecuencia imagen) depende de que exista la misma impedancia en las cuatro fases del mezclador. La impedancia de entrada sube hasta los 5 kΩ para todas las entradas. Debemos remarcar la resistencia de la carga de 4,7 kΩ en la entrada no inversora, a diferencia de la resistencia de 10 kΩ en la entrada inversora. Esto es correcto, ya que la señal que pasa a través de dicha entrada es atenuada con una antifase exacta por la realimentación inversa, lo que divide la resistencia de entrada hasta los 5 kΩ. De este modo, ambas entradas ofrecen la misma resistencia de entrada, que era lo que se pretendía.
Los condensadores de 2,2 nF, junto con la resistencia interna del mezclador y la resistencia serie de 100 Ω, forman un sencillo filtro paso/bajo con una frecuencia de limitación que llega hasta los 100 kHz, de forma que se elimine cualquier vestigio de RF en las etapas de audiofrecuencia. La limitación de frecuencia está bastante por encima del rango de frecuencia de transferencia, lo que significa que las tolerancias del condensador no producen ningún error de fase apreciable. Incluso podemos utilizar un condensador cerámico de disco. Las tolerancias de entre un 10% y un 20% no son ningún problema con cualquiera de los condensadores que están en el camino de la señal y que actúan como elementos de paso/alto con una frecuencia de limitación de unos 300 Hz.
La etapa final tiene una ganancia (dividida en 10 partes) de 20 dB, la cual, sin embargo, puede reducirse a la ganancia unidad por los conmutadores analógicos. Se proporcionan un total de tres pasos de atenuación: 0, - 10 y - 20 dB. Para evitar que este control de ganancia sea demasiado duro, la ganancia también puede reducirse en el propio programa. Como la entrada del receptor presenta una alta resistencia, con lo que podría verse saturada, el atenuador se coloca en la etapa final, de manera que se evite una saturación de la salida. Esto se corresponde con un control de ganancia en un amplificador de FI.

Montaje

La placa de circuito impreso que se muestra en la **Figura 2** utiliza componentes lo más estándar posible, con la excepción de los circuitos integrados LSI (Large Scale Integration, es decir, gran escala de integración) FT 232RL y CY27EE16, los cuales, por desgracia, solamente están disponibles en encapsulado con formato SSOP, con un espaciado entre terminales de 0,65 mm. La **Figura 3** muestra la placa de circuito impreso (PCI) prototipo de nuestro laboratorio, con todos los componentes montados.
La mejor manera de comenzar es soldando los dos componentes de montaje superficial (SMD) en su lugar correspondiente. Se recomienda empezar con los cuatro terminales de las esquinas

antes de soldar el resto de ellos. El estaño que sobre puede retirarse fácilmente con una trencilla para desoldar, revisando a continuación con una buena lupa que no haya ningún cortocircuito que nos dé una desagradable sorpresa.
Los componentes con terminales pasantes no deben presentar ninguna dificultad. El circuito no dispone de ningún componente de RF especial o de puntos de prueba. En un principio no debemos montar los condensadores C12 y C13, ya que el circuito integrado CY27EE16 dispone de unos condensadores internos que pueden configurarse y que nos permitirán alcanzar una frecuencia de exactamente 10 MHz sin ninguna dificultad. Los condensadores C12 y C13 solamente serán necesarios si el cristal que estamos utilizando requiere una capacidad de carga mayor.
Una vez montados todos los componentes es necesario examinar todo el circuito con un multímetro, comprobando que no haya ningún cortocircuito alrededor de las conexiones USB, para evitar dañar nuestro ordenador.

Conexión y alineación

Antes de conectar la primera vez el receptor al puerto USB del ordenador, necesitamos instalar el programa controlador para el circuito integrado FT 232R. Podemos encontrar dicho controlador en la página web del fabricante (www.ftdichip.com/FTDrivers.htm) o, como alternativa, en la sección de programas de descarga de este artículo. La instalación del controlador mediante el programa CDM_Setup.exe elimina automáticamente cualquier resto de viejos controladores FTDI de nuestro ordenador. Una vez que hayamos hecho esto, Windows encontrará de forma automática el controlador correcto en cuanto conectemos el receptor. El mismo proceso proporciona al ordenador, de manera automática, un puerto COM virtual adicional, para lo que no necesitamos conocer qué número de puerto COM ha sido asignado al dispositivo, ya que el equipo configura su propia conexión directa con el FT 232R. El fichero FTD2XX.dll controla las ocho líneas de datos del circuito integrado como si se tratase de un puerto paralelo, eliminando al mismo tiempo cualquier problema de temporización. Para ahorrar tiempo, los múltiples cambios de nivel involucrados en el control del bus I2C son convenientemente almacenados en un “buffer” y, posteriormente, llevados a las líneas de datos de forma ordenada. El programa ElektorSDR.exe nos permite controlar todas las funciones del receptor (vea la Figura 4). Dicho fichero se encuentra en el archivo de descarga como un fichero ejecutable, junto con el código fuente para Delphi. También está disponible para su descarga un documento complementario, en formato pdf, que describe la inicialización y la puesta en funcionamiento del equipo.

Programa decodificador

Prácticamente todas las características significativas del receptor vienen determinadas por la configuración realizada en el programa decodificador de nuestro ordenador. Como indica nuestra investigación [1], disponemos de una colección de programas diferentes entre los que elegir. Por ejemplo, podemos realizar nuestras primeras pruebas con el programa SDRadio [2]. Después de esto descubriremos posibilidades adicionales con los programas DREAM [3] y G8JCFSDR [4].
Cualquiera que sea el programa que hayamos elegido, es de vital importancia configurar la tarjeta de sonido correctamente (esto se describe en el documento complementario). La información sobre los programas se halla en las páginas web correspondientes y en los artículos de Elektor Electronics que se listan más abajo. En la página web del autor de este artículo (wwwb-kainka.de) podemos encontrar consejos adicionales, los cuales también aparecerán más adelante en la página del proyecto en www.elektor–electronics.co.uk y, si fuese necesario, en el foro correspondiente de la misma página web.

(070039-1)

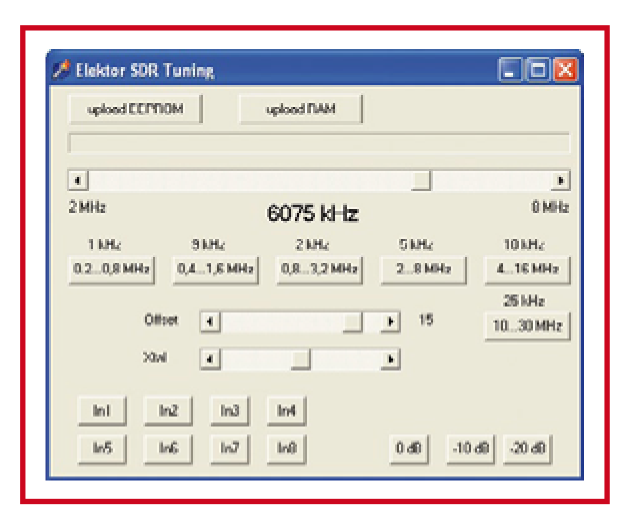


Figura 4. Programa de control de Sintonía SDR de Elektor Electronics.

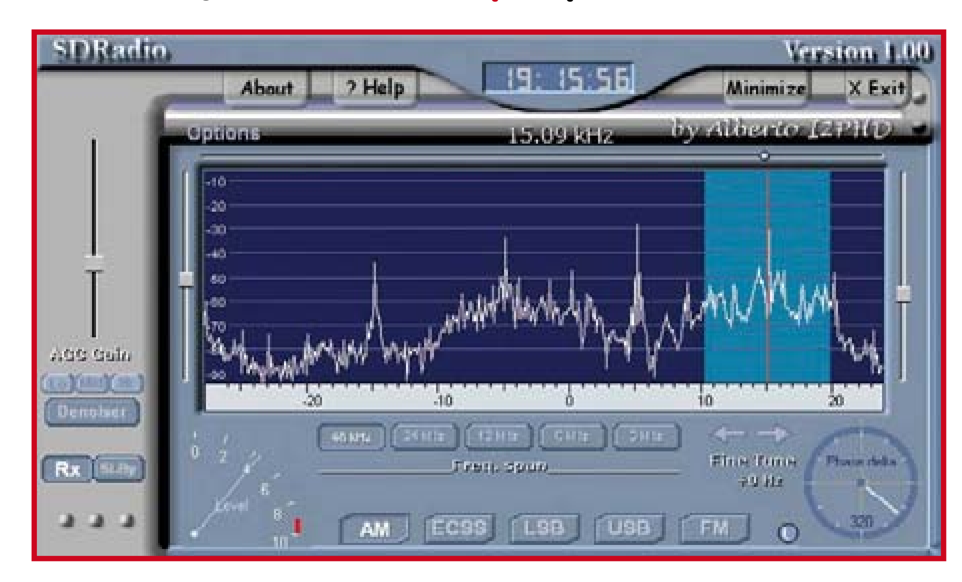


Figura 5. Cuatro estaciones de AM en sintonía dentro del rango del espectro, tal y como se muestra en el programa SDRadio.

Enlaces en Internet:

[1] www.nti-online.de/diraboxsdr.htm
[2] www.sdradio.org/
[3] http://sourceforge.net/projects/drm
[4] www.g8jcf.dyndns.org/

Bibliografía:

Burkhard Kainka: DREAM Team – Programa para Recepción DRM, publicado en Elektor Electronics en abril de 2004.

Wolfgang Hartmann and Burkhard Kainka: ‘Escucha de Radio con Matlab – Programa Diorama para el receptor DRM’, publicado en Elektor Electronics en mayo de 2006.

Burkhard Kainka: I-Q: una aproximación inteligente a la calidad radio, publicado en Elektor Electronics en febrero de 2007.



Profiler

Kit de construcción de una máquina de fresar de interés general

Harry Baggen

¿Alguna vez ha soñado con tener su propia máquina de fresar?, ¿es demasiado cara? Si es así, tenemos la solución perfecta. Hemos trabajado en colaboración con el fabricante belga Colinbus para ofrecer a los lectores un kit que permita la construcción de una máquina de fresar por sólo una parte del coste de un modelo ya fabricado. Esta máquina es adecuada para una amplia variedad de trabajos, que van desde la construcción de piezas para modelos al fresado de placas de circuito.

Tanto los aficionados a la electrónica como los profesionales realizan tareas que exceden a las consideradas estrictamente electrónicas y a menudo tenemos que tratar con muchas tareas mecánicas. Esto incluye llevar a cabo algunos trabajos como realización de placas de circuito impreso, rutado de cables, o diseño de una carcasa adecuada con el correspondiente panel frontal.

Algunos de estos trabajos, si queremos hacerlos nosotros mismos, requieren la utilización de herramientas adecuadas para obtener buenos resultados.

Seguramente muchos aficionados, al igual que diseñadores de prototipos y pequeños laboratorios de desarrollo, harían un buen uso de una máquina de fresar pequeña para este tipo de trabajos. Invertir en esta máquina puede ser rentable para un uso comercial, pero la situación es algo diferente para un uso doméstico. Una máquina de fresar buena y exacta puede costar fácilmente varios miles de euros, incluso aquellos modelos básicos sin todas las funcionalidades (tales como eliminación de limaduras por vaciado y una alta velocidad del eje del motor).

Dedicamos este proyecto especialmente a todos los aficionados de electrónica y profesionales que regularmente continúan su trabajo y/o hobby en casa. La idea para la elaboración de este proyecto surgió espontáneamente durante una conversación con Frank Jacops de la compañía belga Colinbus, especialista en máquinas de fresar.

Cuando hablamos con alguien que no sólo vende máquinas de fresar, sino que también las diseña desde el principio, podemos abordar rápidamente el hecho de que la mayoría de los electrónicos tienen ideas espléndidas sobre todas las cosas que podrían hacer con esta máquina, aunque se encuentren casi siempre con



Especificaciones técnicas de 'Profiler'

Dimensiones	453 (w) x 583 (d) x 468 (h) mm
Área de trabajo máxima	300 (X) x 400 (Y) x 100 (Z) mm
Interface	Serie
Alimentación	240 V, 50 – 60 Hz
Transporte lineal X/Y/Z	guías MultiStab
Driver para X/Y/Z	motores de paso
Velocidad de posicionamiento	60 mm/s
Resolución mecánica	0,0075 mm
Resolución del Software	0,025 mm
Software	Colinbus Interface de Usuario
	Programa de conversión para ficheros
	Gerber y Excellon

el obstáculo del precio. Frank Jacops entendió esto inmediatamente (él ha sido un ávido lector de Elektor Electronics durante muchos años) y sugirió la oferta de un kit de construcción a un precio especial por un tiempo limitado, exclusivamente para los lectores de Elektor Electronics.

Naturalmente, debemos admitir que el kit no es exactamente económico, a 1.679 ó 1.958 más gastos de envío, pero representa un ahorro de casi 1.780 libras, comparado con un modelo ya ensamblado. Esto es una recompensa atractiva para un día de trabajo de ensamblado.

Por supuesto, existen en el mercado pequeñas máquinas de fresar a bajo coste, pero son bastante pequeñas y tienen características limitadas. Si observa las fotos de este artículo, verá que estamos hablando de una categoría diferente. Construcción robusta, alta precisión, y una amplia área de trabajo de 30 x 40 cm (¡ahora sí podemos hincarle el diente!). Esta máquina es adecuada para todo tipo de trabajos, tales como dispensadores, encapsulación, cámara de inspección, fresado de paneles frontales, e incluso para modelar en 3D (utilizando un software separado). Podemos también fresar placas de circuito, aunque en las aclaraciones del fabricante se mencione que este modelo no se ha diseñado inicialmente para este propósito, ya que ello requeriría una mayor exactitud. Sin embargo, nos servirá muy bien, basándonos en nuestra experiencia de PCBs, y el software que se incluye.

El ensamblaje de la máquina es bastante sencillo, y las claras instrucciones escritas prácticamente excluyen cualquier problema. Las placas del circuito con los drivers electrónicos están completamente ensambladas y probadas, así que todo lo que tenemos que hacer es instalarlas y conectarlas.

Diseño

El nombre con el que hemos bautizado a la máquina es 'Elektor Profiler' y es la máquina de fresar más pequeña producida por Colinbus. Su construcción es en gran parte la misma que el modelo comercial CBR-40 (ver www.colinbus.com). La máquina está hecha de partes de acero y aluminio, combinación que proporciona un peso y estabilidad suficiente para soportar el movimiento de la máquina, mientras que la velocidad de recorrido es muy alta gracias al uso de aluminio extrusionado.

El sistema MultiStab emplea tres rodillos en cada extremo, entre barras de acero de precisión presionados entre perfiles de aluminio extrusionado, proporcionando un buen guiado mecánico con un mínimo juego.

El diseño se emplea para los tres ejes. Un rodillo de cada una de las guías se ensambla manualmente para ajustar todo el juego necesario. Cada uno de los tres ejes se maneja con un vástago con una tuerca de recorrido especial.

La máquina de fresar se controla a través de una placa de procesador interior con un Renesas H8/3003 y una placa excitadora con tres circuitos integrados ST L6208, los cuales se encargan de excitar los tres motores de paso. Esta aproximación hace que el tiempo sea independiente del PC conectado. El ordenador simplemente envía comandos y coordina a la placa del procesador, la cual procesa y ejecuta los comandos independientemente.

La potencia de procesamiento del



Figura 1. Las partes mecánicas del kit montaje.

Podemos pedir el kit de construcción para la máquina de fresar Profiler rellenando el documento de pedido en la página web de Elektor Electronics (www.elektor.es). El precio son 1.599 euros, incluido el IVA, más gastos de envío. Los gastos de viaje dependen del país y se encuentran en el documento de pedido en la página web. El kit es suministrado y facturado directa y exclusivamente por el fabricante, no por Elektor. Para ampliar más información visite la web del fabricante: www.colinbus.com/English/cprrange_en.htm

microcontrolador interno no es suficiente para excitar los tres motores al mismo tiempo (este truco se reserva para los modelos más grandes y más caros de Colinbus), así que sólo podemos excitar dos a la vez. La máquina puede ejecutar suaves movimientos en un plano bi-dimensional. Para movimientos en 3D, conmuta rápidamente entre dos ejes, lo que hace que parezca que se mueve rápidamente, debido a que los pasos resultantes desde la interpolación son muy pequeños.

La placa de control en la máquina de fresado tiene una interface serie, pero se puede usar un adaptador estándar USB a RS232 sin ningún problema para manejar la máquina a través del puerto USB del PC.

El eje del motor que se suministra con la máquina de fresar es un modelo básico de Ferm, que nos permite utilizar la máquina inmediatamente. Naturalmente, podemos emplear otros tipos de motores, pero debemos de remarcar que un motor paso a paso profesional de buena calidad puede fácilmente costar más que el kit de montaje entero.

Kit de montaje

¿Qué puedes conseguir por 1.659 ó 2.400 ? Las fotos de la **Figura 1** y **Figura 2** muestran casi todas las partes que se incluyen en el kit. Incluyen todas las partes mecánicas, tornillos y clavijas, rodillos guías, brocas, motores paso a paso, cables etc, además del excitador electrónico de las dos placas de circuito pre-ensambladas. El motor paso a paso Ferm (y soporte asociado), mencionado anteriormente, se incluye también en el kit. Un panel MDF (también incluido en el kit) sirve como plato base.

El software que acompaña a la máquina de fresar consiste en dos programas: una interfase para configurar y operar la máquina y un programa de conversión para circuitos PCBs (ver abajo). El kit incluye instrucciones de ensamblado que proporcionan una descripción paso a paso de cómo montar la máquina.

No hay demasiadas cosas que puedan salir mal durante el ensamblado. Las diferentes piezas se fijan de forma muy exacta gracias a la combina-

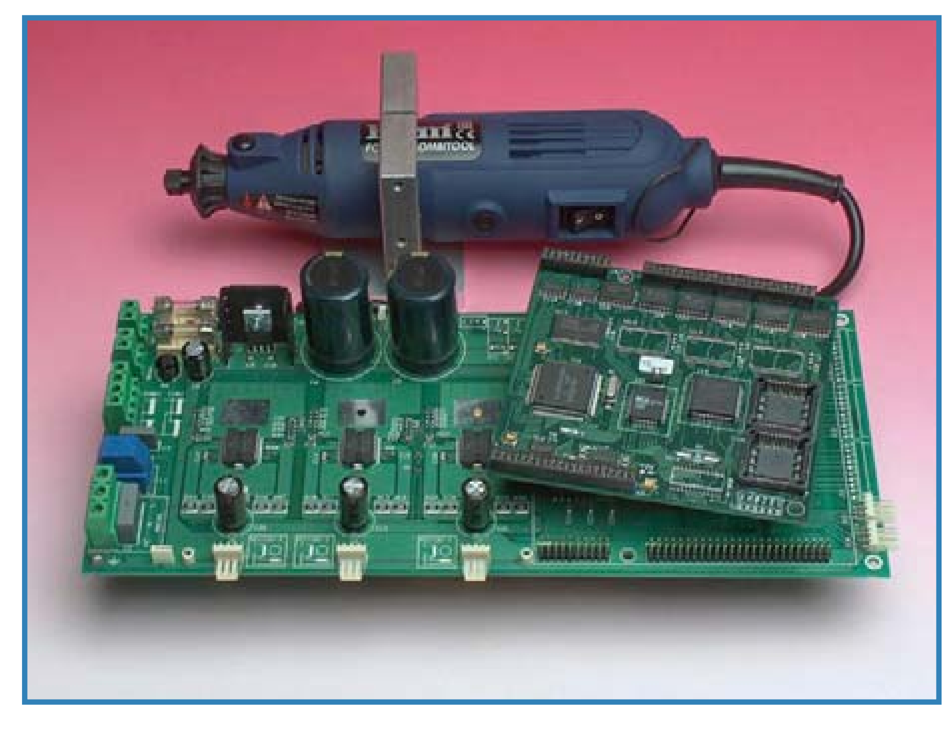


Figura 2. Las placas de circuito ensambladas y el eje y soporte del motor.

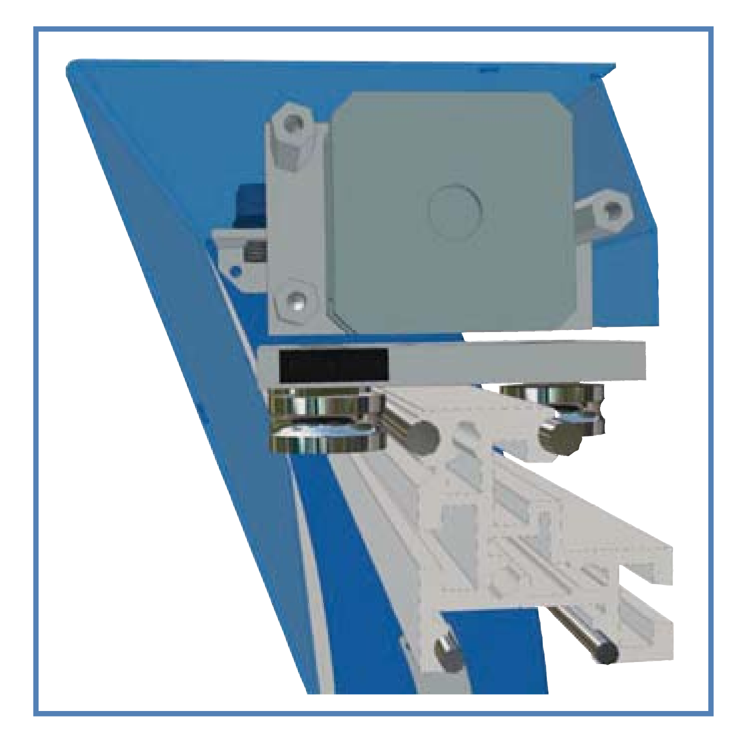




Figura 3. Dibujo de la sección de corte del cabezal montado en el puente.



Figura 4. Detalle de la tuerca de plástico con los pines invertidos, el cual proporciona un acoplamiento sin holguras.

ción de orificios precisos en los paneles de acero y los rodillos de acero en el aluminio extrusionado, por lo que no se necesitan más alineamientos. Debemos de conseguir el cable necesario para las dos placas de circuito para el control y excitadores electrónicos, y soldar en su lugar los conectores y el transformador.

Todos los pasos para el ensamblaje se describen claramente en las instrucciones. Sin embargo, nos gustaría aclarar algunos detalles.

Una vez que ensamblamos parcialmente la máquina, las guías deben de ajustarse.

La precisión de la máquina de fresado depende completamente de este ajuste. Las guías para el puente (en los lados) tienen tres rodillos, de los cuales uno se puede ajustar usando un plato de ajuste. Debemos ajustarlos de forma que haya muy poco juego, pero no se debería dejar demasiado apretado, ya que de otro modo los rodillos se bloquearían. También hay una construcción similar en la parte superior del puente y en la columna del motor, pero la parte posterior debe desensamblarse parcialmente para acceder a los rodillos.

Los drivers usan ejes con soportes trapezoidales que funcionan con tuercas de plástico especiales (ver **Figura 4**). Esas tuercas se fijan con pines transversales que proporcionan el acoplamiento al puente o la columna. Los ejes X e Y también se deben alinear con mucho cuidado para alcanzar la precisión y linealidad especificada. Observe que las tuercas de plástico especiales en esos ejes deben lubricarse usando sólo grasa incluida en el kit. No use grasa normal. El resto del proceso de ensamblado se describe adecuadamente en las instrucciones incluidas con el kit.

Software

La interface de usuario, escrita especialmente para la máquina de fresado de Colimbus, proporciona la interface de operador para la máquina (**Figura 6**). Este programa está diseñado para que incluso usuarios sin experiencia puedan trabajar fácilmente con ella. La pantalla se compone de varias ventanas.



Figura 5. La columna de taladrado y la guía en el puente.

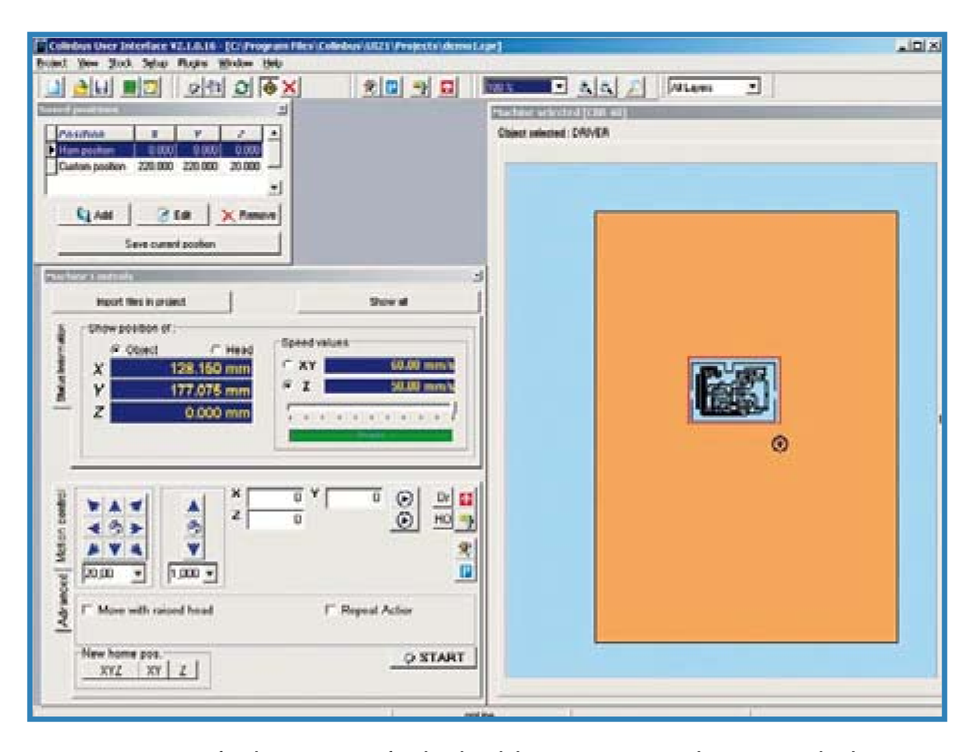


Figura 6. La interface de usuario. La superficie de trabajo de la máquina se muestra en la parte superior derecha.

La superficie de trabajo efectiva de la máquina se muestra en la ventana del lado derecho. Después de cargar un fichero, podemos utilizar el ratón para colocar el objeto a mecanizar en el sitio que deseemos de la superficie.
La máquina empezará a trabajar en la posición exacta. Podemos también especificar los valores de todas las configuraciones y preferencias para la máquina. Además, podemos operar manualmente todos los ejes de la máquina utilizando este programa. Todo lo que podamos pensar puede configurarse, como leer puntos de memoria, puntos de cero relativo, etc, con precisión micrométrica.
La interface de usuario incluye un programa de contorneo de PCBs (**Figura 7**) que puede usarse para editar y mezclar los ficheros Gerber y Excellon generados por los programas más comunes de PCB y después convertir las coordenadas de esos ficheros en contornos usados por la máquina de fresado. Podemos especificar de forma manual los puntos de referencia usados en la placa del circuito y usarlos después para asegurar que los datos en los ficheros Gerber y Excellon son exactamente registrados con todo (el fichero Gerber contiene los datos de las pistas PCB, mientras que el fichero Excellon contiene los datos de orificios). El fichero generado por el programa se puede importar en el programa de interface de usuario de forma que podemos ver la placa PCB, la cual puede arrastrarse a una posición específica en la superficie de trabajo.
Si queremos procesar ficheros en 3D (tales como ficheros DXF), necesitaremos el programa RAMS3D, el cual no se incluye en el kit de montaje debido al precio especial. Finalmente, tenemos algo para los más perfeccionistas: el set completo de comandos para controlar la máquina de fresado está disponible gratuitamente, para que podamos escribir nuestro propio código.

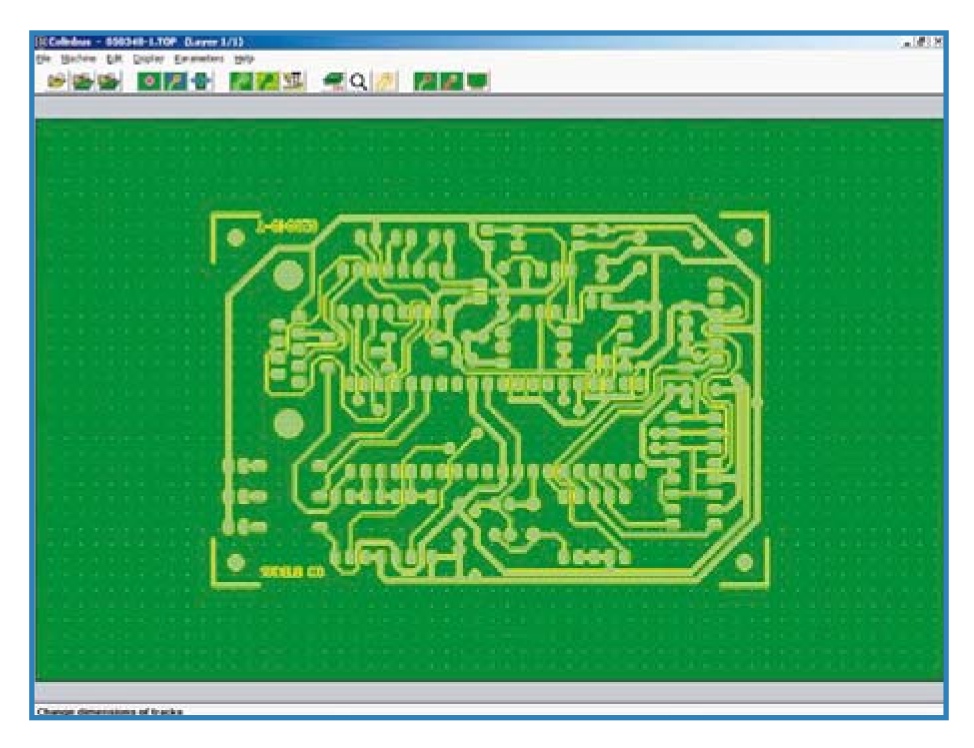


Figura 7. Los ficheros Gerber y Excellon se pueden leer con el programa de diseño de PCBs y convertir en datos para la máquina de fresar.

1001 jobs

Esta robusta y versátil máquina de fresar abre un mundo de oportunidades para aficionados y profesionales que no tengan miedo de remangarse las mangas. Con esto, hacer aberturas rectangulares en una caja no será nada complicado y si necesitamos realizar una ranura para un mando no diremos que nos lo den hecho. Es más, esta máquina de fresar es una herramienta manejable para realizar otras cosas que no tengan nada que ver con el mundo de la electrónica, por ejemplo, podemos utilizarla para realizar las diferentes piezas de un modelo de avión o juguete para nuestro hijo (¡y donar el rompecabezas a un local de caridad!). Resumiendo, se pueden hacer un montón de cosas con esta máquina.
Una vez la hayamos usado, no querremos hacer nada sin ella.

(060232-1)

Tarjeta Flash

Sistema con un 8051 para rápidos desarroll

por Alexander Kniel

Los controladores flash se pueden programar fácilmente, por ello son adecuados para el rápido desarrollo de aplicaciones al igual que para la enseñanza. Hasta ahora, los datos de programas se enviaban la mayoría de las veces a través de la interfaz serie, no obstante, en especial los ordenadores portátiles, a menudo sólo tienen interfaces USB. La solución es nuestra tarjeta Flash de uso versátil. El corazón del sistema es el AT89C5131A, un controlador 8051 ampliado, con núcleo 80C52 y USB de alta velocidad. Además, el dispositivo viene de fábrica con una interfaz de actualización con la que se puede cargar el nuevo firmware. Por otra parte, Atmel suministra también el software adecuado por medio del programa gratuito FLIP.

La "tarjeta Flash" conocida por muchos lectores de Elektor ha acercado el mundo de los microcontroladores a muchos principiantes. Y esto no debe asombrarnos puesto que los controladores que incluyen memoria flash son fáciles de programar, como es el caso del AT89C8252 empleado en aquel entonces. La transferencia de los datos del PC de desarrollo al controlador, con la tarjeta Flash de Elektor, se realiza a través de la interfaz serie, y esta tarjeta, al igual que muchas otras afines, ampliamente utilizada en la enseñanza. Lamentablemente, la buena y antigua "RS232" es cada vez más difícil de encontrar. Además los PC portátiles poseen casi únicamente interfaces USB y ningún puerto de impresora ni interfaz serie. Si un profesor desea entregar tarjetas de experimentación a sus alumnos, para que puedan programarlas en casa con sus ordenadores portátiles, será necesario buscar otra solución.

El autor, profesor de electrónica de una escuela de formación profesional y técnica de Heilbronn, debido a los motivos mencionados, desarrolló una tarjeta Flash que apuesta por un controlador moderno con interfaz USB. Su selección recayó en un AT89C5131AM de Atmel, que posee un núcleo 80C52 y, por lo tanto, al igual que el AT89C8252, pertenece a la familia de los 8051. El componente no sólo contiene una USB de alta velocidad, sino que, más bien, fue desarrollado para su empleo con equipos que se comunican por USB, como impresoras, cámaras, etc. Además el controlador está incluso equipado de fábrica con un interfaz de actualización, a través de la que se puede cargar el nuevo firmware. Exactamente esto le proporcionó al autor, aficionado al hardware y a todo tipo de lenguajes de programación, la oportunidad de construir una tarjeta Flash USB extremadamente sencilla. Atmel suministra también el software adecuado por medio del programa gratuito FLIP. Por tanto, sólo se debe preparar el código, en forma de archivo hex, y ya se puede empezar a trabajar.

Sucesores de la tarjeta flash

La tarjeta desarrollada por Alexander Kniel, en su primera versión, fue montada y puesta en funcionamiento por alumnos en varios PC portátiles. En el laboratorio de Elektor, la placa se modificó un poco; entre otros aspectos, Chris Vossen, técnico de desarrollo de Elektor, integró incluso una interfaz LCD. De esta manera, la tarjeta sigue las huellas de la placa flash de Elektor del año 2001 y es apta para quienes deseen dar los primeros pasos en programación de microcontroladores, pero también para aplicaciones más exigentes en controles de equipos, robótica y otras muchas más.

El corazón de todo el sistema es el ya mencionado AT89C5131AM, un 8051 ampliado. Su núcleo es un 80C52X2 con 6 pulsos de reloj por ciclo de comando. Además de la Flash de 32 KB, el componente posee una RAM ampliado de 1024 Bytes, EEPROM adicional y muchos otros periféricos útiles. También es útil que el controlador, en su versión M, pueda incluso funcionar con 5 V y que se pueda obtener una versión en encapsulado PLCC52 de fácil manejo. No obstante, lo más importante son los módulos de alta velocidad USB-1.1 y 2.0 (para expertos: con punto terminal 0 para Control Transfers al igual que otros seis puntos terminales con memoria FIFO de hasta 512 bytes). Quien desee ocuparse del desarrollo de software USB tiene aquí todo lo que podría desear, no obstante, también se necesita algo de conocimiento técnico. Las demás personas deberán contemplar el controlador como un 8051 normal, que se puede programar a través de un puerto USB.

El diagrama de circuito (**Figura 1**) muestra una alimentación doble, se-

Figura 1. Diagrama de circuito de la tarjeta.

USB

los de software



070125 - 11

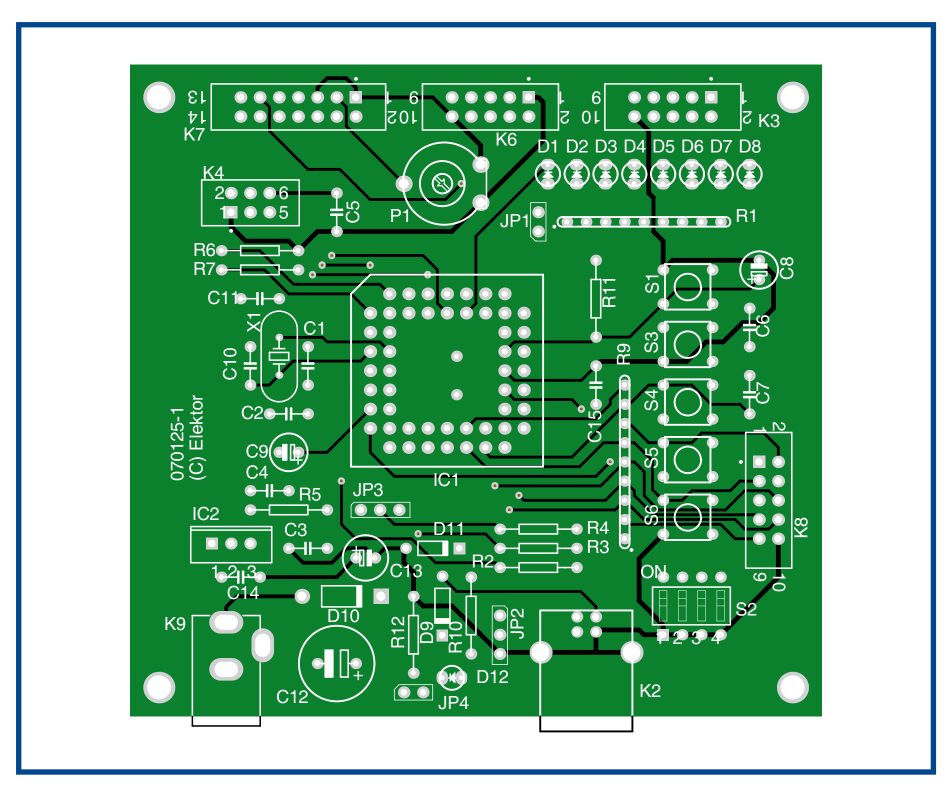


Figura 2. El equipamiento montaje de la tarjeta no debe representar ninguna dificultad.

leccionable a través del USB o (en el caso de que JP4 esté insertado) a través de la conexión de alimentación K9 y del controlador de tensión IC2. Los terminales D+ y D- del controlador son las líneas de datos USB. Para activar el cargador de arranque USB, se debe aplicar un nivel bajo a /PSEN a través de JP2 (el puente que se encuentra hacia el borde de la tarjeta). Si se presiona la tecla S1 de reset se inicia el cargador de arranque con la recepción de datos en la interfaz USB. Para activar el USB, el jumper JP3 debe estar cerrado (puente que se dirige hacia IC2). De esta manera, la resistencia elevadora R4 se encuentra en D+ e indica al PC la presencia de un dispositivo USB de alta velocidad. Quien desee tener mayor comodidad para la conmutación entre el modo de ejecución y el de descarga, puede conectar conmutadores en JP2 y JP3 que se alojen directamente en la tarjeta o que se integren a una placa frontal.

Cuatro puertos completos de 8 bits

El controlador posee cuatro puertos completos de 8 bits, disponibles respectivamente en los terminales o asignados a periferia especial. Se puede acceder al puerto P0 a través de K3 y éste conduce además hacia ocho LEDs, que se pueden conmutar con resistencias en serie con una tensión V_{dd} (+5V). El puerto P1 es de utilización libre y accesible a través de K6. El puerto P2 se encuentra en el terminal K7del LCD. Un LCD opera aquí en modo de 4 bits. También se encuentra disponible el necesario potenciómetro de contraste P1. Finalmente, el puerto P3 está previsto especialmente para introducción de datos y se puede emplear también externamente a través de K8. Para verificar programas propios, la tarjeta está equipada de resistencias elevadoras en el puerto P3 y de cuatro pulsadores (P3.0…P3.3) al igual que de interruptores DIP (P3.4…P3.7). Como es común-habitual, los conmutadores deben tener la propiedad de carecer de rebote, lo cual, la mayoría de las veces, se puede realizar por medio de software. Los terminales P3.2 y P3.3 poseen, no obstante, una eliminación adicional de rebotes por hardware en forma de condensadores (C6 y C7), dado que estos conducen a las entradas de interrupción del controlador. No se debe olvidar el puerto P4 con sus dos terminales P4.0 y P4.1, que forman el bus I2C y que son accesibles a través de K4.

La tarjeta Flash USB (**Figura 2**) se encuentra a disposición en la tienda de Elektor bajo el número 070125-1. Selectivamente, también se puede solicitar de manera directa un juego completo de componentes (070125-71). El montaje sobre la tarjeta no representa ninguna dificultad. ¡En el área del casquillo del USB se debe observar incondicionalmente que no existan conexiones a masa entre D+, D- y 5V del PC! Dado que la tensión de operación de 5 V, D- y D+ del USB no están protegidos especialmente, vale la pena realizar un control con el multímetro – ¡en todo caso, previamente se debe retirar el controlador del zócalo! Debajo del zócalo del CI hay lugar para un condensador cerámico adicional de 100 nF. Previamente, éste ya debe en-

Lista de piezas

Resistencias:
R1 = 1,5k x 8 red de resistencias
R2,R3 = 27 Ω
R4,R12 = 1k5
R5 = 100 Ω
R6,R7,R11 = 4k7
R9 = 10k x 8 red de resistencias
R10 = 1 k
P1 = potenciómetro de 10 k

Condensadores:
C10,C11=22 p
C3 = 2n2
C4 = 10 n
C1,C2,C5..C7,C14..C16 = 100 n
C8 = 10 µ / 16 V
C9 = 1 µ / 16 V
C12 = 1000 µ / 16 V
C13 = 100 µ / 25 V

Semiconductores:
D1..D8,D12 = LED rojo, de baja corriente
D9 = BAT46
D10 = 1N4001
D11 = diodo Zener 6V2
IC1 = AT89C5131AM
IC2 = 7805
X1 = cuarzo de 12 MHz

Además:
JP1,JP4 = regleta de clavijas de 2 polos
JP2,JP3 = regleta de clavijas de 3 polos
K2 = casquillo USB-A
K3,K6,K8 = 2 regletas cóncavas de conectores de 5 pines
K4 = 2 regletas de clavijas de 3 pines
K7 = 2 regletas cóncavas de conectores de 7 pines
K9 = casquillo para fuente de alimentación para montaje de impresora
S1,S3...S6 = teclado en miniatura
Zócalo PLCC
Tarjeta (disponible en la Elektor Shop bajo el número 070125-1).
En Elektor Shop también se encuentra disponible un juego de componentes (070125-71). Consulte el anuncio de la tienda al final de la revista o el sitio www.elektor.de

contrarse montado. De esta manera se tiene un desacoplamiento óptimo de la tensión de alimentación.

Puesta en marcha

Para la primera prueba se debería emplear una fuente de alimentación (8-12 V DC). La alimentación se conecta insertando el puente JP4. Ahora se debe iluminar el LED DI. En el caso de que ya se encuentre conectado un LCD, éste deberá mostrar píxeles oscuros en la línea superior. Eventualmente se debe ajustar el potenciómetro de contraste hasta que ambas líneas se diferencien claramente. Únicamente después del inicio por medio de un programa la línea superior también se vuelve clara. Quien posea un osciloscopio también podrá verificar la señal de 12 MHz en el cuarzo. De esta manera queda en claro que el controlador funciona.

Para la primera verdadera prueba de software se debe cargar un programa. Para esto sirve el FLIP (FLexible In-System Programmer), que se puede descargar gratuitamente de la página Web de Atmel (www.atmel.com). Con la palabra de búsqueda "Flip" actualmente se encuentra la versión FLIP 2.4.6 para Windows (4 MB, revisión 2.4.6, actualizada en mayo de 2006). El documento flip-2_4_6.zip se debe extraer en un directorio propio para que se pueda ejecutar el archivo Setup.exe que allí se encuentra. Siga las instrucciones y acepte las condiciones de licencia y la ruta de instalación propuesta. A continuación, obtendrá una pequeña lista de instrucciones con los pasos de que debe realizar a continuación (**Figura 3**). El programa en sí mismo se encuentra en C:\Programme\Atmel\FLIP 2.4.6\.
Enchufe ahora el conector USB. Coloque el puente JP2 en la posición USB (borde de la tarjeta). Para mayor seguridad, presione además el pulsador de reset S1 e inserte el puente JP3. De esta manera comienza la descarga del firmware por USB y se espera establecer contacto con el PC. El programa muestra las informaciones "Vendor-ID 03EB" y "Produkt-ID 2FFD", con lo que Windows puede determinar cuál es el controlador adecuado. Windows reconoce la presencia de un dispositivo nuevo y pregunta por el controlador correcto. Seleccione el controlador de la carpeta C:\Programme\Atmel\FLIP 2.4.6\usb (véase la **Figura 4**). Una vez que esté instalado, podrá encontrar el dispositivo nuevo en el gestor de Windows. Se lo reconoce por el nombre "Jungo AT89C5130/AT89C5131".

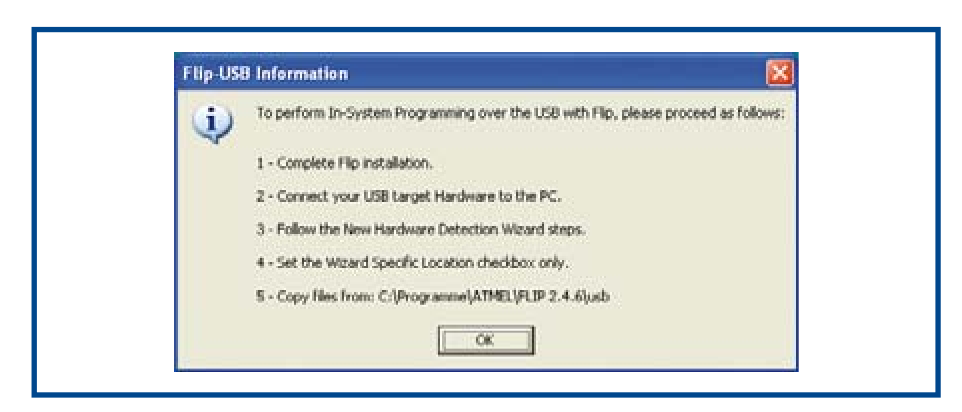


Figura 3. FLIP, el software programador gratuito, enumera los pasos a seguir durante la puesta en funcionamiento de la tarjeta.



Figura 4. Windows reconoce el controlador como un dispositivo nuevo.



Figura 5. Tras un clic en el botón RUN, FLIP carga el programa en la memoria flash del controlador.

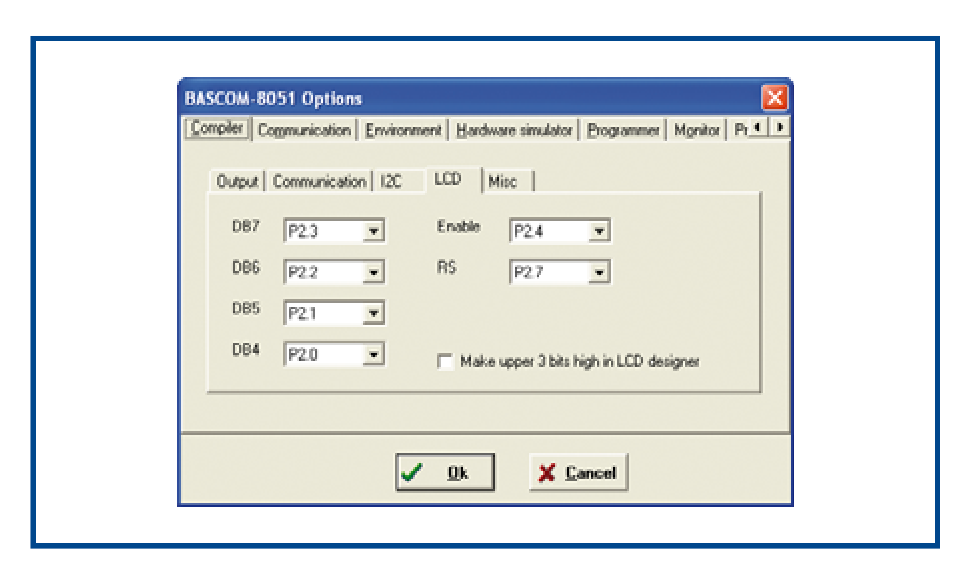


Figura 6. El menú principal del compilador BASCOM.

En el caso de que esto no haya funcionado, se inicia la búsqueda del problema. Las posibles causas del fallo pueden encontrarse en puentes conectados incorrectamente. Si usted, por ejemplo, activa la conexión del puerto USB con JP3 (resistencia elevadora en D+), pero no ha iniciado el firmware interno (JP2 se encuentra todavía dirigido hacia RUN, o no se realizó un reset después de la permutación), Windows indica la presencia de un dispositivo nuevo pero defectuoso. Por el contrario, usted pudo haber iniciado correctamente el firmware de actualización, pero colocó de manera errónea el puente JP3. En este caso, Windows no reconoce que se ha conectado un dispositivo y, correspondientemente, no se establece ningún contacto. No obstante, después de una breve familiarización, los ajustes ya no representarán ningún problema y usted establecerá el contacto con el PC exactamente cuando lo requiera.

Actualización de programa

Inicie ahora el FLIP. Por medio del menú Dispositivo/Seleccionar, o bien con la tecla F2, o el símbolo de CI, se debe selecciona el chip AT89C5131 correcto. Por medio de Ajustes/Comunicación/USB, o bien con el símbolo de cable, o la tecla F3, seleccione la interfaz USB y ábrala. Finalmente, debe cargar todavía un archivo hex adecuado en el menú Archivo/Cargar archivo HEX, o empleando la tecla F4. Seleccione el programa 5131_TEST_ELEKTOR.HEX, que encontrará en la página Web de Elektor, conjuntamente con el texto fuente BASCOM-AVR. Un clic sobre el botón RUN (ver **Figura 5**) carga finalmente este programa a la memoria flash. Para ponerlo en marcha, a continuación, debe cambiar JP2 y presionar la tecla de reset. Atención: En los nuevos controladores el campo "BLJB" se encuentra activado automáticamente. Durante la primera descarga de programa debe eliminar la marca de selección pues de lo contrario el programa no se ejecuta al finalizar la descarga.

Si tras la prueba, desea descargar un nuevo archivo hex, debe retirar el conector del puerto USB y volver a realizar la conexión, naturalmente, con la correcta posición de los puentes y después del reset. A continuación debe volver a abrir la conexión en FLIP. De manera alternativa, se puede mantener insertado el conector USB y retirar únicamente JP3, con lo que el dispositivo se encuentra de igual manera aislado del puerto USB pero continúa recibiendo alimentación de tensión. Para una nueva actualización de programa se debe volver a cambiar en primer lugar JP2. A continuación se vuelve a presionar la tecla de reset y, finalmente, se conecta JP3 dos segundos más tarde. De esta manera, se inicia el dispositivo USB. En FLIP se debe volver a abrir la interfaz y entonces puede comenzar la transmisión de datos.

Test de hardware en Bascom-51

```
,Test sencillo para entradas,
,salidas y LCD
,*********************

Dim X As Byte

P1 = 0

Cls
Lcd „   8051-Test      „
Wait 1
Lowerline
Lcd „    Elektor       „
Wait 3

For X = 1 To 13
Shiftlcd Right
Waitms 200
Next

Cls
Lcd „ Test Port 0      „
Lowerline
Lcd „ Bit 2 exp 0      „
P0 = &B11111110
Wait 1
Lowerline
Lcd „ Bit 2 exp 1      „
P0 = &B11111101
Wait 1
Lowerline
Lcd „ Bit 2 exp 2      „
P0 = &B11111011
Wait 1
Lowerline
Lcd „ Bit 2 exp 3      „
P0 = &B11110111
Wait 1
Lowerline
Lcd „ Bit 2 exp 4      „
P0 = &B11101111
Wait 1
Lowerline
Lcd „ Bit 2 exp 5      „
P0 = &B11011111
Wait 1
Lowerline
Lcd „ Bit 2 exp 6      „
P0 = &B10111111
Wait 1
Lowerline
Lcd „ Bit 2 exp 7      „
P0 = &B01111111
Wait 1
Lowerline
Lcd „ All Bits         „
P0 = &B00000000
Wait 1

Cls
Lcd „ Test Port 3      „
Lowerline
Lcd „=> Port 0 (LED)     „
Wait 3

Status:
P0 = P3
X = P0
Cls
Lcd „     Inputs       „
Lowerline
Lcd „Port 3 =    „ ; X ; „      „
Waitms 60
Goto Status

End
```

Programación con bascom

Cuando se trata de dar los primeros pasos en la programación del sistema, la herramienta correcta es el Basic-Compiler BASCOM-51 (naturalmente, de igual manera, se puede programar el controlador en lenguaje C o en Assembler). Es posible descargar una versión de demostración de este software en la página del fabricante MCS-Electronics (www.mcselec.com). La versión gratuita genera code de hasta 4 KB, lo cual es suficiente para muchas aplicaciones.
La **Figura 6** muestra el menú principal del compilador. Para el control correcto de la tarjeta, se debe establecer en el menú de opciones la asignación de los pines del LCD en el puerto P2 (**Figura 7**).

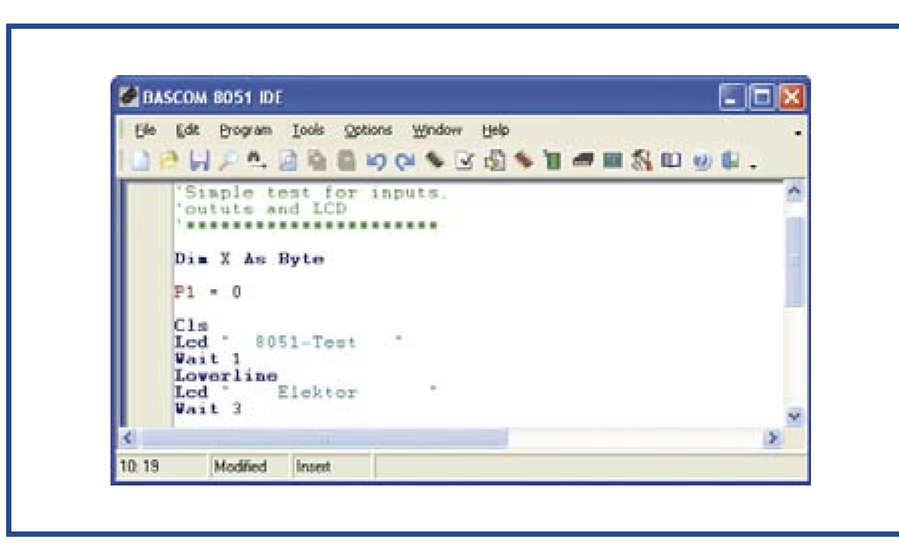


Figura 7. En "Opciones" se debe establecer la asignación de los pines del LCD en el puerto P2.

Bascom permite el ajuste de diferentes archivos de registro para los derivados individuales del 8051. A saber, no existe ningún ajuste especial para el AT89C5131, pero el controlador es ampliamente compatible con el 8052.
Por lo tanto, se debería trabajar con el archivo de registro 8052.DAT.
La lista de instrucciones muestra el programa de prueba en texto fuente. Es de fácil lectura y se explica en gran medida por sí mismo. Tras una introducción al LCD, sigue un bucle de iluminación secuencial para comprobar el funcionamiento de todos los LED del puerto P0. A continuación, en un bucle sin fin, se leen las entradas del puerto y se copian sus estados en la salida P0 y se los muestra también en el LCD. También se puede activar el teclado de ratón S2 y los pulsadores de S3 a S6 para comprobar la asignación de los pines de puerto. De esta manera se ilumina en cada ocasión el LED correspondiente a la salida. Así, el programa sirve también como un verificador casi completo de todo el hardware.

Un par de ideas

Deseamos finalizar exponiendo un par de ideas para desarrollos ulteriores: El controlador posee una memoria EEPROM interna, similar a la que se encuentra en el 89S8252 y en el 89S8253. No obstante, aquí la responsabilidad recae sobre otros registros de control (SFR). Por esta razón, para el control del hardware adicional, no es posible eludir un estudio detallado de la ficha técnica.
Al igual que el 8052, también el AT89C5131 posee incluso una interfaz serie, que se puede dirigir en Bascom con comandos del tipo "print" e "input". Dado que, no obstante, la tarjeta flash USB no contiene una interfaz serie, se debe conectar adicionalmente un controlador de línea (por ejemplo, MAX232). Entonces ya no existen obstáculos contra las aplicaciones típicas de interfaz, esto es, el controlador se puede emplear, por ejemplo, como instrumento de medición de PC; contador o controlador de motor.
Y, por supuesto, el AT89C5131 puede hacer mucho más, esto es, implementar un dispositivo USB completo. En la página Web de Atmel se encuentran algunas notas de aplicación, con sus correspondientes textos fuente, en las que se describe cómo es esto posible. El archivo c5131-usb-kbd-stand-alone-1_0_2.zip demuestra la construcción de un teclado USB. Con el controlador USB y el extenso archivo de software se tiene en principio todo lo que se necesita para el desarrollo de aplicaciones USB propias.

(070999)

Los secretos del I²C

Un analizador de bus I²C para satisfacer vuestra curiosidad

Por Etienne Boyer

En este artículo os presentamos un valioso instrumento, prácticamente indispensable, cuando se trata de analizar lo que sucede en el bus I²C. De hecho, este equipo permite examinar las señales más interesantes que transitan por este bus de interconexión, muy extendido y al mismo tiempo, muy sencillo en sí mismo.

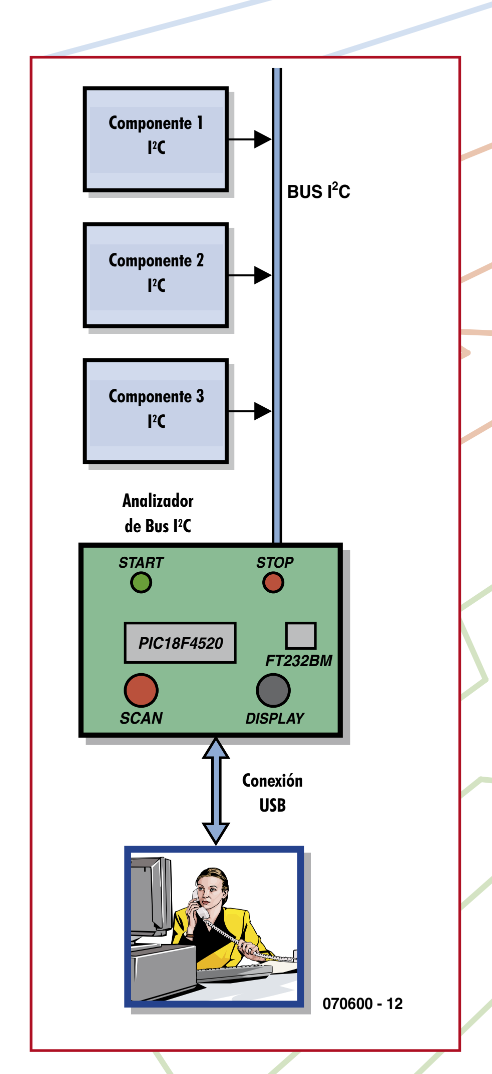


Figura 1. Diagrama de bloques del Analizador de bus I²C en su configuración más sencilla.

El Analizador de bus I²C descrito en el presente artículo se conecta sobre el bus I²C de una aplicación cualquiera, con el fin de extraer, para poder examinar, las señales que transitan en su interior: las señales INICIO, PARADA, DIRECCIONES, DATOS y ADQUISICION más en particular.

El equipo nos podrá servir para la puesta en funcionamiento de una aplicación propietaria que nos esté dando algún problema o para descifrar el funcionamiento de aplicaciones ya existentes (*Ingeniería Inversa*).

El equipo se comunica con un ordenador a través de un enlace USB, configurado como puerto serie virtual (COMx), que se alimenta directamente por el propio puerto USB y que evita el tener que recurrir al uso de un adaptador de tensión de red externo para alimentar el equipo.

El esquema eléctrico

La **Figura 1** nos presenta el diagrama de bloques del circuito. Como ya he comentado, el Analizador de bus I²C se coloca entre el dispositivo que queremos examinar, por medio de la conexión de las líneas del bus I²C de dicho equipo, y el ordenador, que realizará la función de "investigador". Este analizador se puede conectar a un máximo de tres módulos I²C. El corazón y la inteligencia del montaje están reunidos en un microcontrolador PIC, un PIC18F4520, en el que el enlace USB se realiza por medio de un circuito integrado clásico, un FT232BM [1] de la casa FTDI. Se trata de circuito que nuestros lectores ya han encontrado en numerosos montajes descritos en Elektor, con poco que hayan tenido algo de relación con el entorno USB.

La electrónica

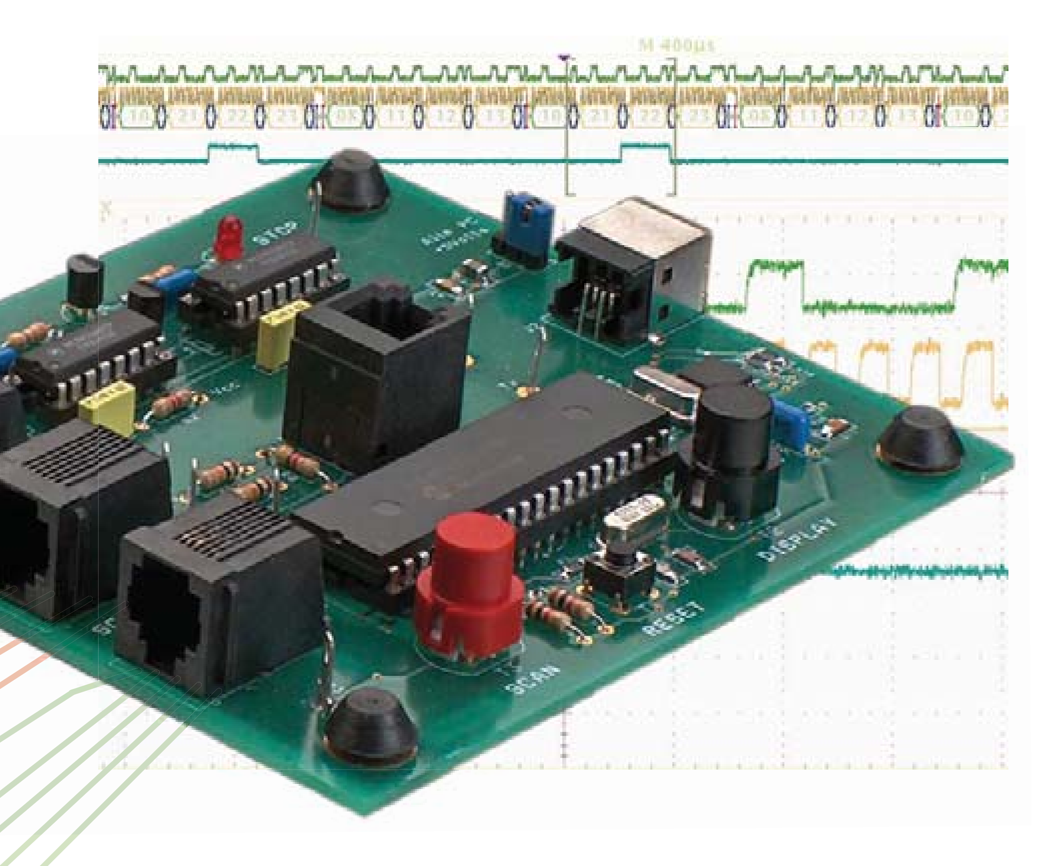
Antes de introducirnos más de lleno sobre el esquema eléctrico, nos ha parecido interesante realizar algunos comentarios sobre ciertos puntos específicos del circuito en la forma de…

…Pequeñas puntualizaciones

El cuadro de texto "Los secretos del I²C y de su bus" recuerda los elementos que constituyen las especificaciones más importantes del bus I²C.

Pero volvamos sobre nuestro circuito. En el bus I²C, el procedimiento de INICIO (START) consiste en detectar un flanco negativo de la señal SDA cuando la señal SCL se encuentra a nivel alto ("1"), un trabajo que realiza correctamente el monoestable IC2.A del circuito 4538, montado según el esquema eléctrico representado en la **Figura 2**.

La duración del monoestable nos da, teniendo en cuenta el retardo de R2xC2, un impulso de 8,2 μs, el cual materializa fielmente el procedimiento de inicio del bus I²C. Esta duración es también compatible con el tiempo de reacción de un microcontrolador, pero incompatible con la persistencia de la retina del ojo huma-



Características técnicas

- Análisis del bus I²C a 100 y 400 Kbits/s
- Registro de 620 eventos I²C contiguos
- Detección física de las señales START y STOP. Visualización sobre 2 diodos LED.
- Comunicación USB por puerto Com virtual.
- Auto-alimentación de 5 V por puerto USB.
- Programación del PIC en lenguaje C (compilador CCS).
- Programa Interfaz Windows hombre-maquina en C++ Builder V5 (Borland)

no: para poder visualizar con un diodo LED la presencia de una fase de Inicio es indispensable el uso de un segundo monoestable. Así, el circuito integrado IC2.B tiene como función la de alargar en unos 150 ms el tiempo de activación del diodo LED verde D2.
Nos encontramos con la misma situación para el procedimiento de PARADA (STOP), esta vez con la ayuda de los monoestables del circuito IC1 y del diodo LED rojo D1. El diagrama de tiempos de la **Figura 3** representa las dos señales de INICIO y PARADA (START y STOP) creadas de esta manera.
Puntualización: los dos pulsos no tienen la misma duración ya que el pulso de INICIO acaba de manera prematura debido a las subidas de la señal SCL. El pulso de INICIO tiene una duración en la práctica de 3 μs, mientras que el de PARADA es más largo que el que se obtiene de manera teórica (8,2 μs) ya que el circuito no trabaja en la banda de tiempo recomendada (100 μs – 1 segundo).

El examen del esquema eléctrico de la **Figura 4** muestra que la electrónica utilizada en este montaje es bastante poca: un PIC, un circuito interfaz USB de la casa FTDI, y un par de circuitos integrados que componen el doble monoestable, los 4538. Examinemos un poco más de cerca sus funciones.
El corazón del montaje toma la forma de un microcontrolador de la casa Microchip, un PIC18F4520 [2], una versión mejorada del ya famoso 18F452. Estos dos microcontroladores son compatibles, terminal a terminal, entre ellos pero también con el célebre 16F877. Su función consiste en analizar las señales de entrada RC3 y RC4 del PIC, atacadas directamente por las señales SCL (***S****erial* ***CL****ock, es decir, Reloj Serie*) y SDA (***S****erial* ***D****a**ta, es decir, Datos Serie*). Es el botón pulsador S3 (SCAN) el que autoriza el inicio del análisis.
Un estudio más detallado del esquema eléctrico no lleva inevitablemente a hacernos una pregunta: ¿el microcontrolador tiene tiempo de poder analizar lo que pasa en el bus I²C?
La respuesta es la siguiente: la duración de la instrucción es de 0,2 μs, lo que representa alrededor de 12 veces la duración de 1 bit (la velocidad máxima normalizada es de 400 Kbits/s) sin embargo, el margen de maniobra es suficientemente fiable para efectuar el tratamiento completo. Además, el reconocimiento de los procedimientos de START y de STOP pasa por la detección de los flancos de las señales, lo que complica el programa de aplicación. El microcontrolador no tiene tiempo de analizar detenidamente el bus I²C. Así pues, se impone una solución física para respaldar y apoyar la solución lógica, gracias a los monoestables sensibles a la llegada de un flanco de señal (como hemos mencionado anteriormente en el apartado de "Pequeñas puntualizaciones").

Ahora que ya hemos realizado la "reparación" del proceso de INICIO (START), basta con que analicemos cada impulso de reloj (señal SCL) y comenzar a muestrear los datos entrantes en la señal SDA en ese instante. Una vez que hemos analizado las señales, el microcontrolador almacenará cada evento en memoria como lo hace un sistema de adquisición de datos (*data logger*). Los cuatro eventos a tratar son el INICIO (START), el OCTETO de datos, la ACEPTACIÓN y la PARADA (STOP). La memoria se rellena al ritmo que el tráfico en el bus y, una vez llena, se transfiere al ordenador por medio de un enlace serie (*el bus USB es, por principio, un enlace serie y no uno paralelo*).
Si el tráfico sobre el bus es demasiado lento, el pulsador S2 (DISPLAY) permite vaciar la memoria sobre el ordenador con el fin de visualizar el resultado sin tener que esperar el llenado completo de la pila.
La comunicación hacia el ordenador se realiza por medio de un circuito bastan-

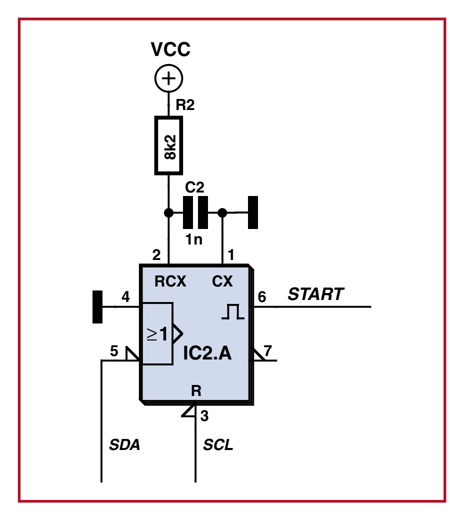


Figura 2. Un monoestable disparado por flanco negativo.

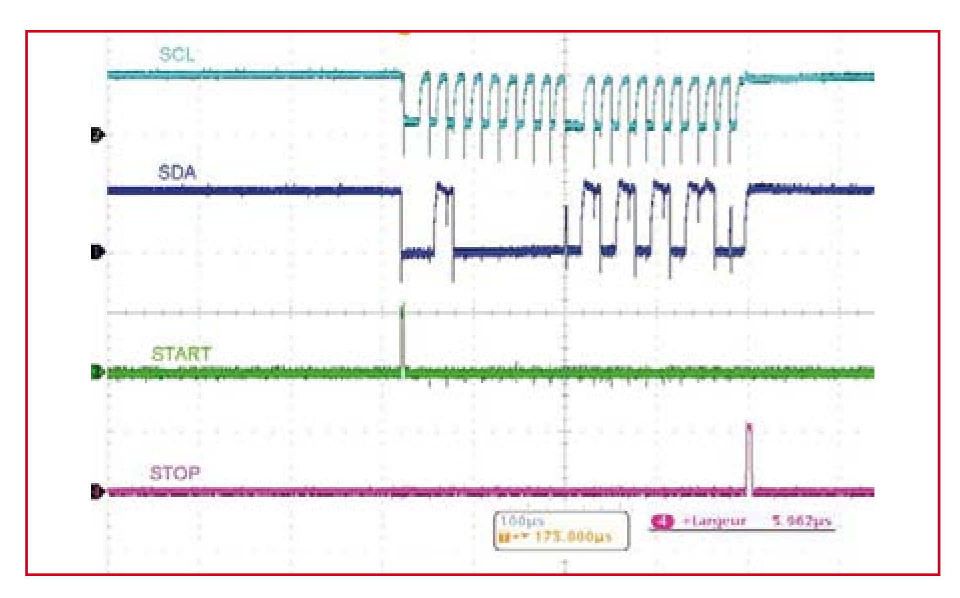


Figura 3. Creacion del Start y del Stop.

te clásico, un FT232 de la casa FTDI, el cual utiliza el bus USB en el modo CDC (***C**ommunication **D**evice **C**lass, es decir, Clase de Comunicación de Dispositivo*). El resto de componentes que rodean el circuito son poco numerosos y se trata esencialmente de un cristal de cuarzo de 6 MHz y de sus dos condensadores acólitos C7 y C8, y del conector USB de tipo B de 4 terminales.

Un par de diodos LED, de color rojo y verde, están asociados a los sucesos de STOP (D1) y START (D2) respectivamente.

Dos resistencias de 27 Ω protegen las líneas Data+ y Data–. El circuito FT232BM conmuta los +5 V sobre la resistencia R16, lo que significa dos cosas desde el punto de vista del puerto USB (PC): por un lado, la presencia de un periférico y por otro, el reconocimiento

Figura 4. La electrónica del esquema eléctrico se resumen en unos pocos componentes, la pareja PIC/conector USB, rodeados de una serie de conectores de todo tipo …

de modo "Full Speed" (Velocidad Completa), ya que la resistencia R16 está conectada sobre la línea Data+.
Un componente interesante es la inductancia L1, con un núcleo de ferrita destinado a suprimir las interferencias de alta frecuencia. De hecho, su impedancia pasa de 0,15 Ω en continua a 70 Ω a 100 MHz, disipando en forma de calor las interferencias electromagnéticas (EMI). Dicha bobina sirve también, como complemento, para realizar la función de fusible, si tenemos la mala idea de cortocircuitar la tensión de +5 V de nuestro montaje, protegiendo así la alimentación del lado del ordenador Para evitar posibles daños, debemos hacer notar la presencia del puente JP1, que permite desacoplar la alimentación del PC.

El circuito impreso

No es necesario decir que un montaje de este tipo merece un circuito impreso. En la **Figura 5** presentamos la serigrafía de la implantación de componentes. La primera etapa, que es también la más delicada, consiste en la soldadura del único circuito integrado SMD (IC4) utilizado en este proyecto. Como siempre sucede en el caso de los componentes SMD (Componente de Montaje Superficial), su soldadura exige una total atención y una mano segura. Los "pads" donde va colocado el encapsulado LQFP-32 del circuito integrado IC4, han sido alargados para facilitar la soldadura con soldador manual de este componente. Comenzamos mostrando el componente a su huella en la placa, respetando su orientación (la referencia redonda da la posición de la patilla 1, y se encuentra directamente a su izquierda). En la serigrafía, la posición de la patilla 1 viene identificada por un pequeño "1". Iniciaremos el proceso de soldadura tomando dos terminales opuestos de manera que, si los otros terminales se encuentran correctamente alineados con respecto a sus "islas" de soldadura, podremos hacer rápidamente la soldadura con la ayuda de un soldador de punta fina y con estaño muy fino. Verificaremos, con una lente de aumento, la calidad técnica de las soldaduras y la ausencia de cortocircuitos a este nivel. Una vez verificado esto, podremos pasar a la colocación del resto de los componentes SMD con encapsulado 1206; operación que irá seguida por la implantación de los pequeños componentes que atraviesan la placa, como resistencias, cristal de cuarzo, condensadores, diodos LED y transistores (atención a la polaridad de todos ellos). A continuación se podrá acometer la colocación de los zócalos (de buena calidad) para los circuitos integrados IC1, IC2 e IC3, terminando con la implantación de los diferentes conectores del tipo RJ-11 (K1, K3, K4 y K5) y los del tipo USB B (K2).
Ahora sólo nos queda colocar los diferentes pulsadores (S2 y S3) y el interruptor de reinicio (S1).
Importante: si la alimentación se hace por medio del puerto USB del analizador, lo que suele ser el caso habitual, tendremos que colocar el puente correspondiente en la posición "on" del conector JP1.
Después de una última revisión de todo el montaje para asegurarnos de la ausencia de errores o de cortocircuitos, es el momento de interconectar la placa de nuestro proyecto al ordenador por medio de un cable USB. Así, podremos verificar la presencia de la tensión de alimentación en los puntos previstos de los zócalos, operación que debemos hacer

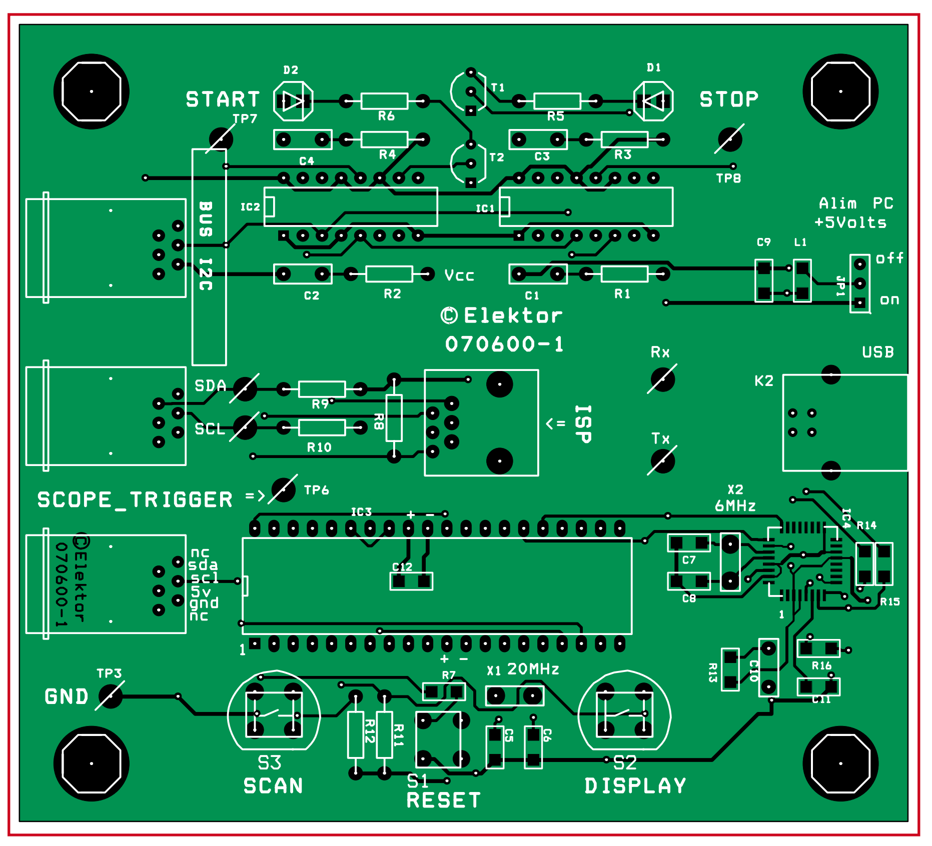


Figura 5. Serigrafía de la implantación de componentes del Analizador de bus I²C.

Lista de materiales

Resistencias:
R1,R2 = 8kΩ2
R3,R4 = 1MΩ5
R5,R6 = 330 Ω
R7,R13 = 470 Ω
R8 = 4kΩ7
R9,R10 = 10 kΩ
R11,R12 = 1 kΩ
R14,R15 = 27 Ω
R16 = 1kΩ5

Condensadores:
C1,C2 = 1 nF
C3,C4,C10 = 100 nF
C5 à C8 = 22 pF
C9 = 10 nF
C11,C12 = 33 nF

Semiconductores:
D1 = LED 3 mm rojo
D2 = LED 3 mm verde
T1,T2 = 2N7000
IC1,IC2 = 4538
IC3 = PIC18F4520 (programado **EPS070600-41**)
IC4 = FT232BM (FTDI)

Varios:
K1 = conector RJ-11 de terminales (vertical)
K3 a K5 = conector RJ-11 de 6 terminales (horizontal)
K2 = conector USB macho tipo B
L1 = núcleo de ferrita
X1 = cristal de cuarzo de 20 MHz (HC 49/4H)
X2 = cristal de cuarzo de 6 MHz (HC 49/4H)
S1 = tecla contacto miniatura
S2,S3 = tecla contacto D6 (roja y negra)
JP1 = conector "header" de tres terminales macho SIL + puente
Placa de Circuito Impreso 070600-1 disponible en las direcciones habituales
Ficheros del diseño de la placa **070600-1** + programa (programa + fichero .hex) **070600-11** disponibles para su programación en www.elektor.es

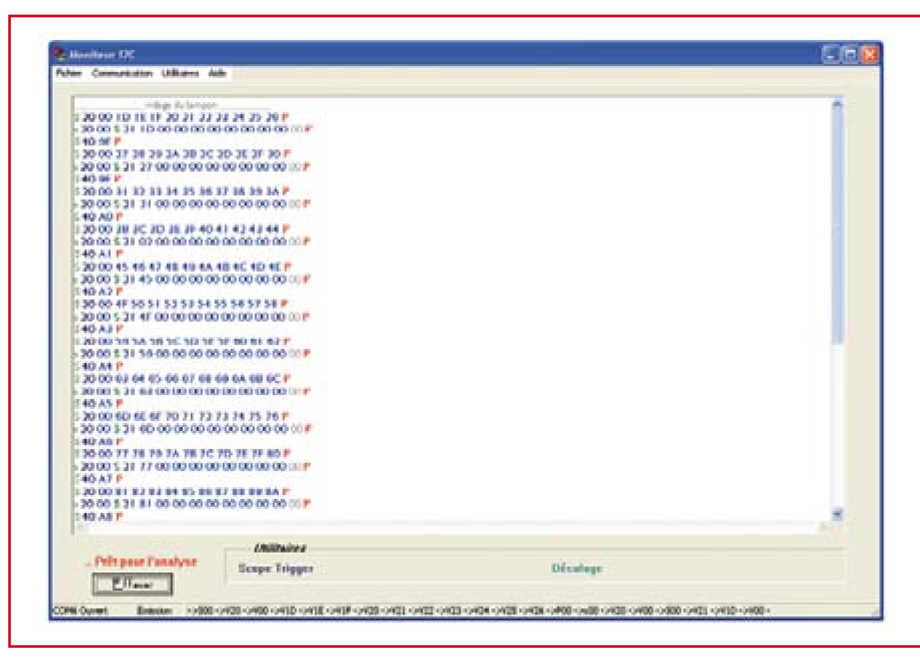


Figura 6. Captura de pantalla del programa Monitor I²C.

con la ayuda de un multímetro. Si todo está correcto, podremos desconectar el analizador e implantar los últimos circuitos integrados (IC1 a IC3), cuidando de no equivocar su posición. La placa dispone de un cierto número de puntos de prueba: TP1 a TP8 en la serigrafía, de los cuales, las líneas SCL, SDA, Rx y Tx, se corresponden respectivamente con los puntos de prueba TP1, TP2, TP4 y TP5). Estos puntos de prueba podrán ser dotados, o no, con un terminal como el que dispone el prototipo.
Para las pruebas, el prototipo puede alimentarse con 5 V a partir del conector del bus I²C o por hilos soldados directamente sobre el circuito.
La verificación de la parte eléctrica viene facilitada por el hecho de que los monoestables son independientes del programa instalado: en el momento en que el prototipo se conecta sobre un bus I²C existente, los cuatro monoestables deben reaccionar a la llegada de los correspondientes pulsos de START y de STOP, realizando el encendido de los diodos LED verde y rojo, respectivamente.
Seguidamente, el PIC puede ser ya programado.

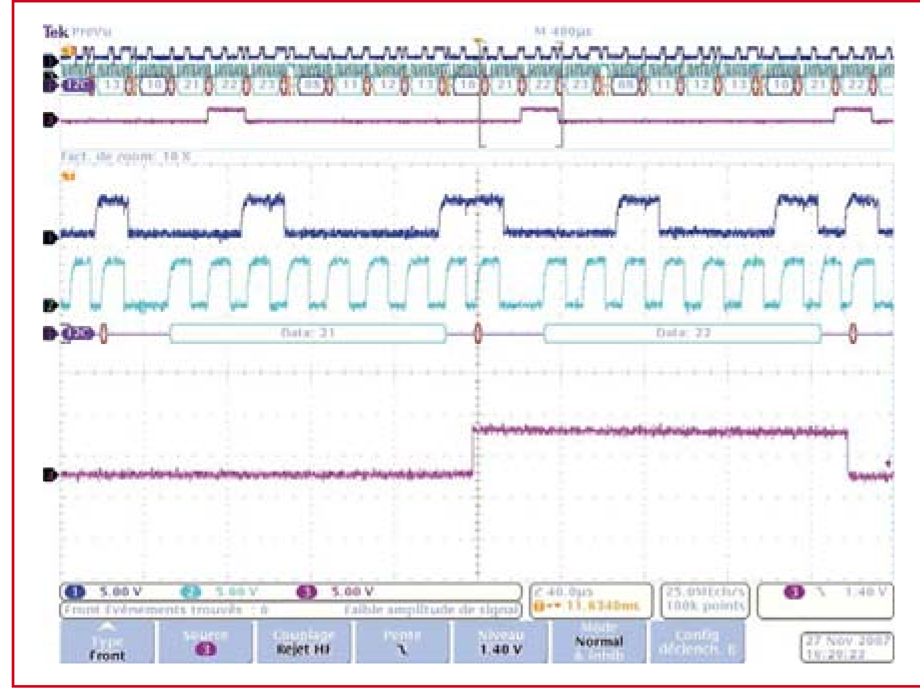


Figura 7. Captura de pantalla del disparo en un osciloscopio

La parte de programación

El código fuente del programa del microcontrolador está escrito en lenguaje C y ha sido compilado con la ayuda del compilador cruzado PCH Compiler V4.010 de la casa CCS (Custom Computer Services). Este compilador, que tolera ciertas variaciones sobre el lenguaje C "académico", se encuentra así muy bien adaptado a la programación para la electrónica.
El desarrollo se ha realizado sobre MPLAB V.7.62. El programa, que se ejecuta bajo Windows, está escrito en C++ Builder V5 (Borland). El programa para Windows también se puede compilar bajo CodeGear 11, el último IDE de Borland disponible en versión de evaluación de 30 días. Ya conocen todo lo necesario nuestros lectores.
En el centro del prototipo hay un conector K1, ISP, pensado con el fin de permitir la puesta en funcionamiento (ICD = ***In-Circuit Debugging**, es decir, Depuración en el Circuito*) y la programación sobre el prototipo (ISP = ***In System Programming**, es decir, Programación en el Sistema*) del microcontrolador.

Una ojeada al programa

Si la electrónica es sencilla, el programa colocado en el interior del microcontrolador debe ser igual de potente. Se propone a nuestros lectores la telecarga del dispositivo desde nuestra página web (www.elektor.es) bajo la denominación de EPS070600-11. Abordemos algunos aspectos prácticos.

El interfaz hombre-máquina: Instalación

Esta aplicación, escrita en lenguaje C++ Builder V5.0, corre bajo Windows y se instala muy fácilmente con la copia del ejecutable **moniteur_I2C.exe**. La aplicación necesita la instalación previa de los controladores (*drivers*) en el PC. Para hacer esta tarea es interesante consultar la página web de FTDI [3] y los artículos de Elektor precedentes que tratan este tema.

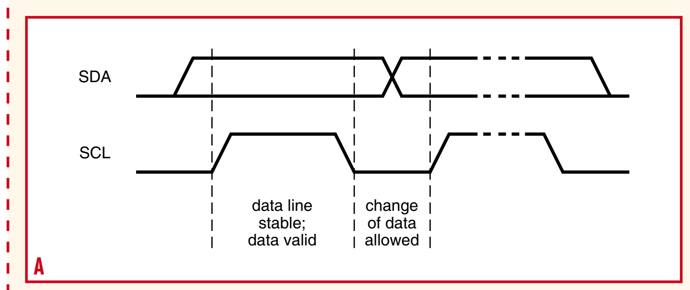
Configuración RS-232

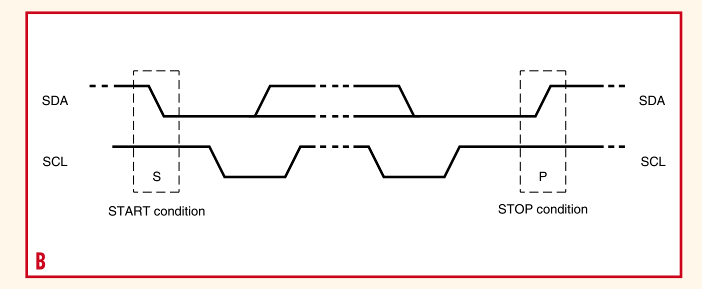
Al lanzar la aplicación, un primer cuadro de diálogo permite elegir el puerto COM Virtual por el que se va a realizar el enlace USB y se van a recibir los datos que el PIC envía bajo la forma de un enlace serie asíncrono clásico (velocidad de transferencia de 128.000 baudios).
La barra de estado indica la apertura correcta del puerto serie.

Presentación de los resultados

El análisis de los datos se lanza por la

Los secretos del I²C y de su bus





Es a Philips, en la época en que este fabricante de equipos de audio formaba parte del pelotón de cabeza, a quien debemos la presencia del bus I²C (Inter Integrated Circuit). Los principales dominios de aplicación de este bus han sido la domótica y la electrónica doméstica al principio de los años 80. Fue en esta época en la que hizo su aparición el microprocesador en los televisores y era necesario encontrar una técnica, realizable y sencilla, para interconectar los diferentes subconjuntos electrónicos de un aparato de este tipo.

El bus I²C es un bus serie síncrono que tan sólo utiliza tres líneas: Datos, (SDA), Reloj (SCL) y Masa (que sirve de referencia).

El funcionamiento del bus se basa sobre las nociones del maestro (el periférico que gestiona la comunicación, genera el reloj y emiten los datos) y el esclavo (periférico que recibe los datos y envía un acuse de recibo por una señal de reconocimiento). No debemos equivocarnos ya que, a pesar de parecer bastante rústico, éste bus puede acomodarse a muchos microcontroladores sin que se produzca conflicto, con la condición de respetar ciertas reglas.

Las cuatro situaciones más importantes de este protocolo se muestran a continuación:

1) Transferencia de un bit sobre el bus I²C (***Figura A***).

El reloj no tiene la “consistencia” de un reloj verdadero, sabiendo que puede tener duraciones de tiempo variables.

2) Condiciones de START y STOP (***Figura B***).

En el inicio de la comunicación, la línea SDA pasa a “0” mientras que la línea SCL permanece a “1”. Es el “StartBit” (Bit de Inicio). Al final de la comunicación, cuando la línea SDA pasa a nivel “1” y la línea SCL permanece a nivel “1” también, nos encontramos en presencia del “StopBit” (Bit de Parada).

3) transferencia de datos sobre el bus I²C (***Figura C***).

Cuando esta condición está definida, el dispositivo maestro coloca el bit de mayor peso sobre la línea SDA. Y valida el dato forzando brevemente la línea SCL a nivel alto. Y continúa este modo de funcionamiento para los diferentes bits hasta alcanzar el bit de menor peso. Cuando la transmisión ha finalizado, el dispositivo esclavo fuerza la línea SDA a nivel bajo. Esta es la…

4) Señal de aceptación (acknowledge) en el bus I²C (***Figura D***).

Es el dispositivo esclavo el que emite esta señal para indicar la recepción de todos los datos. Si todo está OK, fuerza la línea de datos a “0”.

A pesar de su sencillez, el bus I²C permite el tratamiento de operaciones relativamente complejas. Los diferentes diagramas de tiempo tomados en este artículo ilustran claramente este funcionamiento.

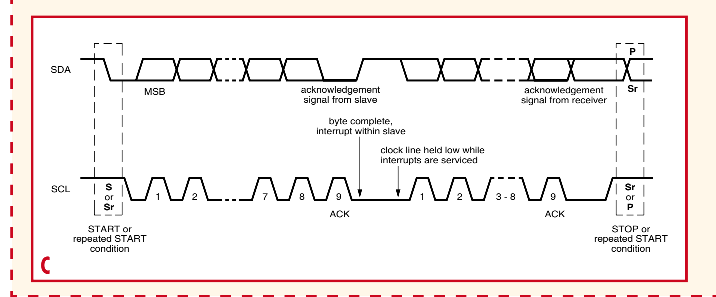
Debemos señalar que un dispositivo maestro puede también recibir datos desde un esclavo (maestro receptor).

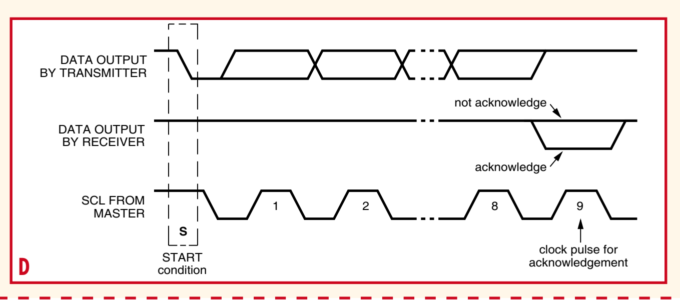
Si nuestros lectores quieren saber más sobre el tema, pueden dirigirse a la dirección de Internet:

http://www.nxp.com/acrobat_download/literature/9398/39340011.pdf (inglés).

Allí se pueden obtener las Especificaciones de Enero de 2000 sobre el bus I²C, con las que podran conocer todos los detalles.

(Fuente de las imágenes: Philips Semiconductors)





pulsación sobre el pulsador S3 (SCAN). Los eventos recibidos por el bus I²C son registrados por la placa y aparecen siguiendo un código de colores correspondientes a los eventos en el bus I²C. la captura de pantalla de la **Figura 6** representa la pantalla principal cuando la adquisición ha finalizado, esto permite reconocer los bits de START en verde y los de STOP en rojo, las direcciones en azul marinó y los datos en azul rey. En cualquier caso, el valor del bit de aceptación va a modificar esa bella apariencia. La presencia del bit de aceptación conserva los bellos tonos azules, pero la ausencia de dicho bit de aceptación va a convertirlos en gris.

En la barra de estado también aparece el formato de los códigos trasmitidos sobre el enlace serie entre el dispositivo de análisis y el PC (en código ASCII).

Ejemplo:

-> S00 para el START
-> V20 para un octeto adquirido cuyo valor es 20(HEX)
-> v20 para un octeto no adquirido cuyo valor es 20(HEX)
-> P00 para el STOP.

Función sincronización del Osciloscopio

Es una función muy útil que permite configurar en el analizador 1 octeto de sincronización que, cuando está presente en la trama, dispara un impulso de sincronización sobre el terminal SCOPE_TRIGGER a fin de sincronizar un osciloscopio en modo “disparo externo”. Esto permite visualizar la forma de onda de las señales SDA y SCL justo en ese instante. Por ejemplo, la captura de pantalla de la **Figura 7** representa la señal SCOPE_TRIGGER para el octeto de sincronización 21(HEX). La parte alta de

los datos registrados muestra claramente que, a cada llegada de este octeto, la señal báscula durante la duración del octeto siguiente. La parte mediana se corresponde con una lupa (zoom) con un factor de zoom de 10 y hace aparecer la transición de la señal SCOPE_TRIGGER en el momento de la llegada del noveno impulso de la señal de reloj SCL. Este instante, que se corresponden normalmente con la respuesta del receptor, nos permite leer un nivel lógico "1" sobre la señal SDA, que viene a testimoniar la ausencia de la aceptación (ACK): el circuito I^2C de dirección 10(HEX) al que iban destinados los datos 21 22 23, es así considerado ausente del bus o defectuoso.

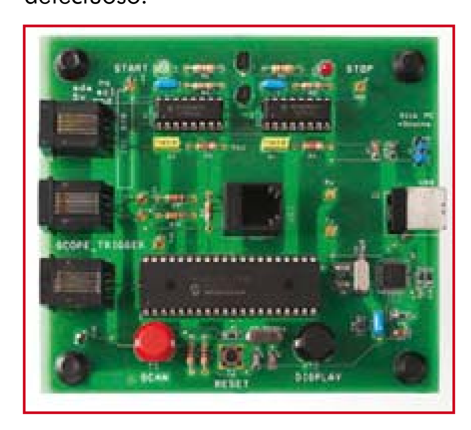


Foto del prototipo. Se nota la evolución de la denominación de los componentes con respecto a la versión final.

Función Desplazamiento
Otra función bastante útil permite declarar el inicio del registro de un cierto número de eventos. Así pues, estos eventos son sustituidos por un punto en el momento de su presentación en pantalla.

Resumen del I^2C
Una página de ayuda integrada comprende un limitado resumen de algunas definiciones de los eventos presentes en un bus I^2C. además, esta ayuda muestra el registro de señales reales.
Para transmitir los datos sobre el bus I^2C es necesario vigilar dos condiciones en particular: la condición de START y la condición de STOP. La condición de START se corresponde con un flanco descendente de la señal SDA, mientras que la señal SCL se encuentra a nivel alto.
La condición de STOP se corresponde con un flanco ascendente de la señal SDA mientras que la señal SCL se encuentra a nivel alto.
A continuación, ocho impulsos generados por el reloj permiten muestrear los 8 bits del octeto recibido, comenzando por él bit de mayor peso.

El noveno impulso de reloj permite una respuesta, es decir una aceptación por parte del componente que recibe el octeto precedente. Si el componente está presente, éste pone la señal SDA a nivel bajo como signo de aceptación. Éste es el principio de protocolo o "apretón de manos". Si no, la línea queda a nivel alto y el emisor del octeto puede reaccionar.

¿Qué queda aún?

¡Poca cosa! Tan sólo queda conectar el bus I^2C de una aplicación al analizador de bus I^2C, por un lado, y conectar éste último al PC por el otro. Seguidamente, lanzamos el programa **monitor_I2C.exe** y esperamos a que aparezca la pantalla con los primeros datos recibidos, tras haber activado previamente el pulsador SCAN, seguido algunos instantes después por el pulsado del conmutador S2, DISPLAY.

Conclusión

Con una gran sencillez de uso, este montaje permite, a partir de componentes clásicos (Circuitos lógicos CMOS, Microcontrolador PIC, interfaz USB), efectuar el análisis de señales presentes sobre un bus I^2C. Una evolución del montaje sería el uso de un PIC que contenga una pila USB (PIC18F4550). Esta solución permite simplificar el material suprimiendo el circuito interfaz USB y el cristal de cuarzo de 6 MHz, mejorar también las prestaciones en velocidad, ya que el PIC 18F4550 genera, gracias a un PLL, un reloj de 48 MHz. El inconveniente reside en el aumento de la complejidad del programa. Las librerías generadas por los editores de los compiladores cruzados en lenguaje C existen y proporcionan numerosos ficheros fuente (Compilador MPLABC18, CCS), pero la puesta a punto global de todo el programa no es algo que se consiga fácilmente.

Esta utilidad práctica nos permitirá ver lo que se pasa sobre el bus I^2C de la central de adquisición de datos descrita en otro artículo de este número, sabiendo que dicha central posee un bus I^2C que interconecta el reloj en tiempo real al resto del sistema. ¡Buena caza!

(070600-I)

El autor



El autor de este artículo ha realizado estudios de ingeniero en el INSA de Lyon y, después, se ha dedicado a la enseñanza, pasando las oposiciones de la Educación Nacional francesa.

Enseña electrónica a los estudiantes de la sección BTS (Brevet de Technicien Supérieur) desde hace bastantes años, a través de los cuales ha vivido y transmitido la evolución fantástica de la tecnología: los componentes discretos (transistor de una unión 2N2646), las memorias, los microprocesadores y la llegada de los ordenadores personales (PC).

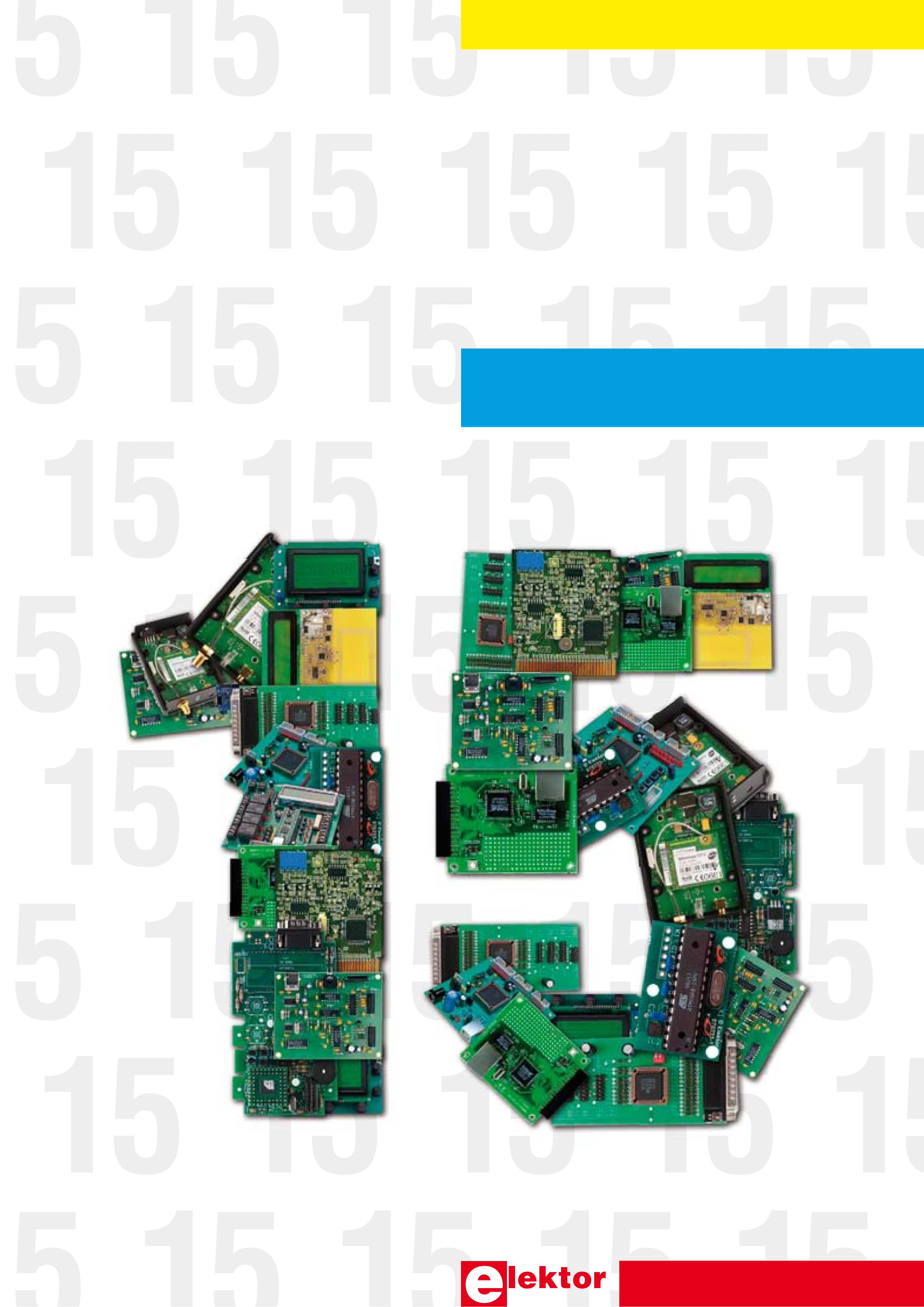
La formación ha evolucionado también apoyándose ahora sobre la comprensión de los sistemas electrónicos complejos: instalación, configuración y reparación.

Incluso ahora, cuando se encuentra delante de un proyecto interesante, le llega la sensación de sentir esta máxima: ¡qué buena profesión la de profesor!

Etienne Boyer

Enlaces en Internet

[1] *Hojas de características del FT232BM*
www.ftdichip.com/Documents/DataSheets/ds232b18.pdf
[2] *Hojas de características del PIC18F4520*
ww1.microchip.com/downloads/en/DeviceDoc/39631a.pdf
[3] *Página web de FTDI*
www.ftdichip.com/
[4] *Compilador CCS*
www.ccsinfo.com/


elektor